UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/512,592	02/23/2000	Bruce M. Dickens	2039-301	3314

7590   10/27/2004

BRUCE DICKENS
DICKENS-SOEDER 2000 LLC
3892 CEDRON STREET
IRVINE, CA 92606

EXAMINER
COBY, FRANTZ

ART UNIT	PAPER NUMBER
2161	41

DATE MAILED: 10/27/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

	Application No.	Applicant(s)
Office Action Summary	09/512,592 ; 90/005,592	DICKENS, BRUCE M.
	Examiner	Art Unit
	Frantz Coby	2171

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _09 June 2004 and 05 October 2004_.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-76_ is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-76_ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☐ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
    Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____ .

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

This is in response to Applicant's RCE (Request for Continuing Examination) filed on October 05, and

Amendment filed on June 09, 2004 in which claims 1-76 are presented for examination.

Claims Status

Claims 1-76 are pending.

New Matter

The amendment filed on June 30, 2004 (Paper # 40E) is objected to under 35 U.S.C. 132

because it introduces new matter into the disclosure. 35 U.S.C. 132 states that no amendment shall

introduce new matter into the disclosure of the invention. The added material, which is not supported

by the original disclosure, is as follows:

```
10      open structure toos:name 'otms src dir:tools'

        open #2 : name `last inv.dat', access output
        print " Tools `Last Inventory Data Format' Check for 1996 Inventory"
        print " Tool No        "; " Model No "; " Last INV " ; "LAST IN'V
        print "====" "," ==== "," ====== "," ======
        print "Extract Data:"

        print #2: "ToolNo "- "Model No "" LAST INV

"LAST INV"
        print #2: " "====" "," ==== "," ====== "," ======
        print #2: "Exact Data"

20          extract structure tools
            yy$ = lpad$ (element$ (tools (last inv 3 "/" 2 "0")
            mm$ ; lpad$ (element$ (tools (last invl 1 "/") 2. °`0")
            dd$ = lpad$ (element$ (tools (last inv) 2 "/") 2 "0")
            cc$': vy$ -+. ·ri mm$ + ddb,
            c 1$ = change$ lc$ ·/. ")
            if cl$f 1:21,'50' then
              c$ = _ '20' ± cl$
            else
              c$='19'+cl$
            end if
!               include c$ , `19960101'
```

```
            sort bar
            tools(model)
    -       sort byrpad$(c$,8, '0')
    !       if c$[1:8], '19960101' then
    _       print tools(toolno); tab (23); tools(model): &
    _                       tab(35) ; toos(last inv); tab(44): c$
            print #2: tools(toolno); tab (23); tools(model). &
                            tab(35); toos(last inv), tab(44):, c$
                if valid   cl$ "digiis"' ; 0 then
                print: tab(53): " Date format is not digits"
                print #12: ;tab(53)-, " Date format is not digits"
                end of
    !           if valid (cl$ "minlength 6") = 0 then
                print: rab(50): " Date format is short"
    !           print #2: :tab(50)i ` Date format is short"
                end if
                if tools(last inv) _ "" then
                print; tab(53)- " Date format is blank-'
                print #2: tab(53)z-" Date format is blank"


                end if
    30      end extract
                print
                print "Sorted Data:"
                rrint
    40      .for each tools
                c 1$ + change$ (tools(last inv `/' "
                print tools(toolno): tab (23); tools(model): &
                tab 135); toolsllast iny),; tab 44 -, c
                    if valid ( cl $. "digits") = 0 then
    !               print; tabl53)i " Date format is not digits"
                    print #2: ;tab(53): " Date format is not digits"
                    end if
    !               if valid ( cl$. "minlcnath 6") = 0 then
    !               print: tab(53); " Date format is short"
                    print #2: iab(53); " Date format is short"
    !               end if
```

Applicant is required to cancel the new matter in the reply to this Office Action.

Reissue Applications

Objection to Oath/Declaration

2a. The reissue oath/declaration filed with this application is defective because it fails

to identify at least one error which is relied upon to support the reissue application.

Further, it fails to refer to the amendment of 10/15/2002. See 37 CFR 1.175(a)(1), 37

CFR 1.63(b)(2), and MPEP 1414.

The new Oath/Declaration submitted on 02/19/2002 is unsigned as required by

MPEP 1401.01.

3. The original patent, or an affidavit or declaration as to loss or inaccessibility of

the original patent, must be received before this reissue application can be allowed. See

37 CFR 1.178.

4. Changes made in the certificate of correction have not been incorporated into the

specification of the reissue application. Applicant is required to submit a substitute

specification, which complies with reissue practice.

5a. This application is objected to under 37 CFR 1.172(a) as lacking the written

consent of all assignees owning an undivided interest in the patent. The consent of the

assignee must be in compliance with 37 CFR 1.172. See MPEP. 1410.01. A proper

assent of the assignee in compliance with 37 CFR 1.172 and 3.73 is required in reply to

this Office action.

5b. This application is objected to under 37 CFR 1.172(a) as the assignee has not

established its ownership interest in the patent for which reissue is being requested. An

assignee must establish its ownership interest in order to support the consent to a reissue

application required by 37 CFR 1.172(a). The submission establishing the ownership

interest of the assignee is informal. There is no indication of record that the party who

signed the submission is an appropriate party to sign on behalf of the assignee. 37 CFR

3.73(b) . A proper submission establishing ownership interest in the patent, pursuant to

37 CFR 1.172(a), is required in response to this action.

5c. The person who signed the submission establishing ownership interest has failed to

state his/her capacity to sign for the corporation or other business entity, and he/she has

not been established as being authorized to act on behalf of the assignee. See MPEP §

. 324.

5d . It would be acceptable for a person, other than a recognized officer, to execute a

submission establishing ownership interest, provided the record for the application

includes a statement that the person is empowered to sign a submission establishing

ownership interest and/or act on behalf of the organization.

5e. Accordingly, a new submission establishing ownership interest, which includes

such a statement above, will be considered to be executed by an appropriate official of

the assignee. A separately filed paper referencing the previously filed submission

establishing ownership interest and containing a proper empowerment statement would

also be acceptable.

6. Claims 1-76 are rejected as being based upon a defective reissue declaration under

35 U.S.C. 251 as set forth above. See 37 CFR 1.175. The nature of the defects) in the

declaration is set forth in the discussion above in this Office action.

Claim Rejections - 35 USC § 112

7. The following is a quotation of the first and second paragraphs of 35 U. S. C. 112:

The specification shall contain a written description of the invention, and of the manner and

process of making and using it, in such full, clear, concise, and exact terms as to enable any

person skilled in the art to which it pertains, or with which it is most nearly connected, to

make and use the same and shall set forth the best mode contemplated by the inventor of

carrying out his invention.

The specification shall conclude with one or more claims particularly pointing out and

distinctly claiming the subject matter which the applicant regards as

his invention.

     Claims 16-67, 69-73, 75- 76 are rejected under 35 U.S.C. 112, first paragraph, as

containing subject matter which was not described in the specification in such a way as to

reasonably convey to one skilled in the relevant art that the inventor(s), at the time the

application was filed, had possession of the claimed invention.

lii.   Claims 1-76 are rejected under 35 U.S.C. 112, second paragraph, as failing to

particularly point out and distinctly claim the subject matter, which the applicant regards as

his invention.

A.   Claims 32 and 69 call for sorting dates "in the form of CIC2YIY2", while the

specification only describes sorting with the format C 1 C2Y 1 Y2M 1 M2D 1 D2. (col.2, lines

15-21, col.3, lines 38-48). These two sorting formats are different from each other since the

former excludes month and date data f7-om the sort keys, resulting in a faster sort, with a

potentially different resultant sequence, than the latter, in which, unlike the latter, the runs of

data for any given century and year combination are not further sorted by month and day.

One of ordinary skill in the art at the time of the invention would have immediately

appreciated that a CCYYMMDD sort is fundamentally different than, as contrasted with

being a species for the genus of, a CCYY sort because of these two dramatic differences in

the sort results. Therefore, because a CCYY sort is not merely a broader genus for the

species of CCYYMMDD, the disclosed CCYYMMDD sort does not implicitly satisfy the

written description and enablement requirements under 35 USC 112, first paragraph.

Therefore, the claimed sorting format is new matter since it is not disclosed in the original

specification.

B.   Claims 33, 60-61, 64-65 and 70 call for reformatting to occur "without changing " or

"without modifying" the symbolic date representations during the reformatting when the

specification merely indicates that the YYMMDD date format is reformatted to appear in the

form CCYYMMDD (col.3, lines 41-43). It is apparent that the original specification is

devoid of any disclosure of how such reformatting is performed "without changing" or

"without modifying" the symbolic date representation. In fact, the suggestion of

reformatting without changing representation is on its face a contradiction, for the reformat is

to change representation. Therefore, the claimed limitation "reformatting to occur "without

changing " or "without modifying"" is new matter because this subject matter was given

neither a written description nor enabling description in the original disclosure.

C.   Claims 16-30, 32, 34-67, 69-71, 75 and 76 call for processing relative to a "pivot date" or

"pivot year" when such terms are nowhere defined or even mentioned in the original

specification. Therefore, the claimed limitation "pivot date" or "pivot year" is new matter

because this subject matter was given neither a written description nor enabling description

in the original disclosure.

D.   Claims 20-21, 62-65 and 71 call for "reformatting" or "storing" "separately" from the

symbolic representations in the database or from the database when the original specification

merely suggests reformatting or sorting the date. However, the original specification does

not disclose such "separate" reformatting or storing. Therefore, the claimed limitation of

"separate storing " or "separate reformatting" is new matter because this subject matter was

given neither a written description nor enabling description in the original disclosure.

E.   Claims 16-25, 31-33, 66-67 and 72 call for "collectively further processing" when the

specification makes no mention of such "collective" further processing. Similarly, claims

36-43 call for "collectively sorting" or "collectively manipulating" when the original

specification merely suggests sorting and manipulating. However, it does not mention such

"collectively" sorting or manipulating. Similarly, claims 34-61, 63 and 65 call for the step of

"running a program collectively" when the original specification, perhaps, only implicitly

discloses the "running of the program". However, such "collective" running of the program,

is not disclosed. Therefore, the claimed limitations of "collective processing", "collective

sorting", "collective manipulating" or "collective running" are new matter because this

subject matter was given neither a written description nor enabling description in the original

disclosure.

F. Claims 36-37, 40-41, 48-49, 51-59, and 69 call for the running of a program after a sorting

operation has been performed. However, the original specification does not provide a written

description of such running of a program subsequent the step of sorting. Similarly, claims 3839, 42-43

call for data manipulation before running of the program. No written description is provided for such

data manipulation before running the program in the original specification. Therefore, such limitations

are new matter because this subject matter was given neither a written description nor enabling

description in the original disclosure.

G.   Claims 46-59 call for "repeating the step of converting at least a substantial portion" of the

specified data. The original specification does not disclose the conversion of such substantial portion.

Therefore, such limitation is new matter because this subject matter was given neither a written

description nor enabling description in the original disclosure.

H.   Claims 34-65 and 70-71 call for "converting " symbolic representations "by windowing the

symbolic representation" when the specification merely discloses the selection of a 10 decade

window. The verb "windowing" appears nowhere in the specification, and its meaning is unclear.

Therefore, such limitation is new matter because this subject matter was given neither a written

description nor enabling description in the original disclosure.

I.   Claims 35, 37, 39, 41, 43, 45, 49, 51, 53, 55, 57 and 59 call for the step of "opening the

database prior to the step of converting" when the original specification makes no mention of

opening the database. Therefore, such limitation is new matter because this subject matter

was given neither a written description nor enabling description in the original disclosure.

J.   Claims 34-65, 70 and 71 call for the avoidance of an "ambiguity" by reformatting or

converting date representation. The original specification merely suggests that dates

containing only two digit year representation, and without reformatting, may sort improperly.

It does not mention or discuss any such claimed ambiguity. Therefore, such limitation is new

matter because this subject matter was given neither a written description nor enabling

description in the original disclosure.

K.   Claims 1-15, 31, 33, 68, 72-74 call for the selection of a "YAYB value for the first

decade" of a window. There is no known meaning for the "value of a decade" and the

original specification is devoid of any description of what the "value of a decade" is.

Because this subject matter was in the original disclosure, such limitation is not new matter.

However, it is rejected under the second paragraph of 35 USC 112 because the meaning of

the claim phraseology is so devoid as to be wholly indefinite.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or

described as set forth in section 102 of this title, if the differences between the subject matter

sought to be patented and the prior art are such that the subject matter as a whole would have

been obvious at the time the invention was made to a person having ordinary skill in the art

to which said subject matter pertains. Patentability shall not be negatived by the manner in

which the invention was made.

9.   Claims 1-3, 5, 7, 9-10 are rejected under 35 U.S. C. 103(a) as obvious over Daniel P.

Shaughnessy, US. Patent No. 5, 630,118, filed on November 21, 1994 and issued on May 13,

1997 (Shaughnessy, hereinafter) in view of Masakazu Hazama, Japanese Application No.

05-027947, published on February 5, 1993 (Hazama, hereinafter).

Because there are so many claims with so many subject matter elements, the detailed,

claim by claim analysis may be too repetitive. In an effort to provide an overriding clarity to

the following rejections, the following are noted:

• The Shaughnessy reference is an essentially complete teaching of the claimed subject matter. In

particular, it teaches modifying those dates that have a two digit identifier less than some

predetermined pivot date, changing the format of the date, and sorting the results. However,

Shaughnessy does not explicitly state that the predetermined pivot date is less than any date in the

database.

• Hazama is provided, consequently, as an explicit teaching of the need for the pivot date to be less

than any date in the database.

• Therefore, it follows that, logically, the suggested process of converting all dates in

the database, wherein two digit dates are converted into four digit dates as taught in

Shaughnessy cannot operate correctly unless the pivot date is less than any date in

the database. This is due to the fact that any dates in the database that were less than

the pivot date would be incorrectly altered to a date in the succeeding century.

It also follows that this assignment of a pivot date is simply a species of the

genus of setting program parameters according the specific input data criteria.

Further, it follows that one of ordinary skill in the art of programming would

know and would be adept at setting parameters to correctly process a set of

data. Applicant is reminded that the software development process consists

both of design, in which process is matched to the scope of the input, and

testing, in which data are entered through the process to check results. Thus,

either of which would have provided sufficient notice to the ordinary skilled

artisan that setting the pivot date to accommodate the input data is a

necessity.

• Although the following claims may be rejected over Shaughnessy alone, given the

logical necessity of setting the pivot date properly, and having articulated and placed

the above facts and analysis in the record consistent with the recent decision of In re

Lee, 61 USPQ2d 1430 (CA FC 2002), the Hazama reference is provided to

demonstrate that, apart from being logically necessary, this attribute of the pivot date

being earlier than the dates in the database was, in fact, known to those of ordinary

skill in the art at the time of the invention, and not a ground breaking discovery by the

applicant.

As to claim 1, Shaughnessy substantially discloses the invention including the claimed

method of processing symbolic representations of dates stored in a database' as a method

and system for modifying and operating a computer system to perform operations on date

fields having a two digit representation for the year without mistaking the year 2000 and the

year 1900 (col.1, lines 7-14 et seq). Shaughnessy also suggests the conversion of all dates

within the database from a two digit format to a four digit format as a viable, but costly

alternative for the year 2000 problem (col. 1, lines 31-46 et seq).

The method as recited in the body of the claim is particularly taught by the cited references

as follows:

Shaughnessy discloses the step of "providing a database with symbolic representations of dates

stored therein according to a format wherein Msub.1I M.sub.2 is the numerical month designator, D.

sub.1 D.sub.2 is the numerical day designator, and Y.sub.1 Y.sub.2 is the numerical year designator'

as a database having a 6 digit-field for storing a Date type in the form of MMDDYY(see appendix in

col. 18, Date Type A), wherein the MM represents the month, DD represents the day and YY

represents the year for a particular six digit-date. Further, Shaughnessy discloses the claimed all of

the symbolic representations of dates falling within a 10 decade period of time' as a date having a

cycle or a range of a 100 years (col. 18, Cycle/Range C1 = THE DATE CYCLE IS 100 YEARS). As

pointed out in column 2, lines 11-14 and column 3, lines 4-8 of Patent No. 5,806,063, all dates in

commercial and industrial databases span within one 100 year. Shaughnessy's system being of the

commercial or industrial kind described in the cited patent, must therefore, as a practical

matter, incorporate this limitation. Shaughnessy also discloses a subroutine for determining the

current date to thereby select a 100 year cycle wherein the current date is the pivot date and wherein

the cycle ends a 100 years from said current date (col.5, lines 31-36). Additionally, Shaughnessy

discloses the step of determining a century designator C.sub.1 C.sub.2 for each symbolic

representation of a date in the database, Csub.1 C.sub.2 having a first value if Y.sub.1 Ysub.2 is less

than Y.sub.A Y.sub. B and having a second value if Y.sub. I Y.sub.2 is equal to or greater than

Y.sub.A Y.sub.B as the comparison of the current date to the date when the system was installed with

the modifications (modified system install date) to thereby determine the century value (col.5, lines

36-65 et seq). Alternatively, Shaughnessy discloses the comparison of the YYMMDD portion' of the

date to the corresponding date portion at the end of the 100 year cycle to thereby determine the

century value (col.7, lines 7-15 et seq). Finally, Shaughnessy discloses the step of reformatting the

symbolic representation of the date with the values C.sub.1 C.sub.2, Y.sub.1 Y.sub.2, M.sub.1

M.sub.2 , and D.sub.1 D.sub.2 to facilitate further processing of the dates' by appending the

determined century value before the YYMMDD date in order to yield a CCYYMMDD date format

(col.5, lines 46-51; co1.6, lines 57-65 et seq) and by returning one date field with the converted date

to the subroutine and a means for returning a parameter to the application program for use in further

operations (col. 1, lines 47-54 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest Y I Y2 year designator in the database in the selection Shaughnessy

specifically suggests that it might be desirable to set the current date to a date which compares low to

all other dates (col. 7, lines 16-17 et seq) of the 10 decade window. However, Hazama discloses an

analogous date processing system having two digit dates spanning from the 20th century to the 21st

century stored in memory (page 2, claim 1) wherein, for a 100 year window (restricting the all the

dates in the database between the 20th and 21st centuries), the pivot date for the window is selected

based on a two digit year date that is smaller than the smallest two digit year date in the database

(e.g. if smallest two digit date stored in database is 73, the pivot date for the 100 year window is

chosen to be 72). (page 4 of translated document, last paragraph). It would have been obvious to one

of ordinary skill in the art of data processing to combine the teachings of the cited references. The

ordinary skilled artisan having read Shaughnessy would immediately see the need to determine

which 100 year span to use. This determination would have led the ordinary skilled artisan to the

Hazama reference, which teaches the pivot date being smaller than the smallest two digit date in the

database having all the dates within a 100 year period as a solution to restrict the selection of

Shaughnessy's window and thereby forcing all dates already stored in the database to fall in the 20th

century.

As to claim 2, Shaughnessy discloses the invention as discussed in the rejection of claim 1, as

well as the claimed limitation whereby a ` 10 decade window includes the decade beginning in the

year 2000' by suggesting the use of a 100 year window that includes a decade date in the 21st

century (col. 6, lines 28-29 et seq).

As to claim 3, Shaughnessy discloses the invention as discussed in the rejection of claim 2, as

well as the claimed limitation whereby, `the step of determining includes the step of

determining the first value as 20 and the second value as 19' by assigning the century value to 19 if

the YYDDD portion of the date is greater than or equal to the corresponding portion of the

corresponding portion of the modified system install date (col. 5, lines 40-46) and by assigning the

century value to 20 if the pivot date is less than the modified system install date (co1.5, lines 52-60 et

seq).

As to claim 5, Shaughnessy discloses the invention as discussed in the rejection of claim 1, as

well as the claimed limitation, wherein `the step of reformatting includes the step of reformatting each

symbolic representation of a date into the format C.sub. I C.sub.2 Y.sub. I Y.sub.2 M.sub. I M.sub.2

D.sub. I D.sub.2' as the conversion of the current date from a six digit format (YYMMDD) into an

8-digit format (CCYYMMDD) (col. 5, lines 48-50 et seq).

As to claim 7, Shaughnessy discloses the invention as discussed in the rejection of claim 1, as

well as the claimed limitation, wherein the step of providing a database includes the step of

converting pre-existing date information having a different format into the format wherein M.sub. I

M.sub.2 is the numerical month designator, D.sub. I D.sub.2 is the numerical day designator and

Y.sub.I Y.sub.2 is the numerical year designator' by as the converting the current date in a six digit

format (YYMMDD), wherein YY represents the year, MM represents the month and DD represents

the day (col. 8, lines 18-27 et seq).

As to claim 9, Shaughnessy discloses the invention as discussed in the rejection of claim 1,

as well as the claimed limitation of `storing the symbolic representation of dates and their associated

information back into the database after the step of reformatting' by saving the converted date in the

database (col. 6, lines 1-3 et seq). Also Shaughnessy col. 1 lines 31-35 and col. 4 lines 12-23 indicate

this is in fact one solution to the Y2K problem, but suggests that it is an expensive solution -

Shaughnessy teaches away only from an economic, not a technical viewpoint. The examiner takes

administrative notice that this solution of storing the symbolic representations back in the database,

again, taught by Shaughnessy as an available solution, is the only permanent solution, and is

therefore inevitable - the economic rationale in Shaughnessy is temporary - eventually the data in a

database spans over 100 years.

As to claim 10, Shaughnessy discloses the invention as discussed in the rejection of claim 9, as well as the claimed limitation of ` manipulating information in the database having the reformatted date information therein' by performing updates on the converted dates and saving said converted dates in the database (col. 6, lines 1-22 et seq).

10.   Claims 4, 6, 8 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view of Hazama, as applied to the rejection of claims 1-3, 5, 7, 9-10 above, further in view of Booth et al., Implementation in Clipper 5A Developer's Guide (Booth, hereinafter).

As to claim 4, Shaughnessy and Hazama disclose the invention as discussed in the rejection of claim 1 above. Shaughnessy and Hazama do not particularly disclose the additional step of `sorting the symbolic representations of dates, after the step of reformatting.' However, Booth discloses an analogous system that utilizes the Clipper programming language to process dates stored in a database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq. In particular, analogously to Shaughnessy, Booth discloses the SET EPOCH command for comparing an entered two digit date with the year digit (pivot date) of the epoch setting to determine the century to place the date into (p. 941, see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a corresponding eight digit date (C1C2Y1Y2M1M2D1D2) (see p. 940-941). Additionally, Booth complements Shaughnessy and Hazama by suggesting the sorting of converted dates after having been reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary skill in the art of data processing at the time of the instant invention to combine the teachings of the cited references. Booth's teaching of sorted reformatted dates would facilitate the Shaughnessy- Hazama's system to return the reformatted dates in chronological sequence. . Such

disclosed sorting would therefore be very useful when indexing the database in date order, as

suggested by Booth on page 945.

As to claim 6, Shaughnessy substantially discloses the invention as discussed in the rejection

of claim 5 above. Shaughnessy and Hazama do not specifically disclose the additional step of

`sorting the symbolic representations of dates using a numerical order sort, after the step of

reformatting.' However, Booth discloses an analogous system that utilizes the Clipper programming

language to process dates stored in a database to thereby derive other dates therefrom (p. 939, lines

1-3 et seq. In particular, analogously to Shaughnessy, Booth discloses the SET EPOCH command for

comparing an entered two digit date with the year digit (pivot date) of the epoch setting to determine

the century to place the date into (p. 941, see SET EPOCH paragraph et seq) thereby converting a

six digit date (MMDDYY) into a corresponding eight digit date (C 1 C2Y I Y2M 1 N12D I D2) (see p.

940-941) . Additionally, Booth complements Shaughnessy and Hazama by suggesting the sorting of

converted dates after having been reformatted by the SET EPOCH command (p. 945). It would have

been obvious to one of ordinary skill in the art of data processing at the time of the instant invention to

combine the teachings of the cited references. Booth's teaching of sorted reformatted dates would

facilitate the Shaughnessy - Hazama's system to return the reformatted dates in chronological

sequence. . Such disclosed sorting would therefore be very useful when indexing the database in

date order, as suggested by Booth on page 945.

As to claim 8, Shaughnessy and Hazama substantially disclose the invention as discussed in

the rejection of claim 1. Shaughnessy and Hazama do not specifically disclose the step of selecting

Y.sub.A Y.sub.B such that Y.sub.B is 0 (zero). However, Booth discloses an analogous system that

utilizes the Clipper programming language to process dates stored in a database to thereby derive

other dates therefrom. See p. 939, lines 1-3 et seq. In particular, analogously to Shaughnessy, Booth

discloses the SET EPOCH command for comparing an entered two digit date with the year digit (pivot

date) of the epoch setting to determine the century to place the date into (p. 941, see SET EPOCH

paragraph et seq) thereby converting a six digit date (MMDDYY) into a corresponding eight digit date

(C 1 C2Y 1 Y2 M 1 M2D 1 D2) (see p. 940-941). Additionally, Booth complements Shaughnessy and

Hazama by suggesting that the pivot date be set to 90 by selecting set epoch to be 1990 (Le. YAYB =

90), such that YB equals to zero (p. 942). It would have been obvious to one of ordinary skill in the art

of data processing at the time of the instant invention to combine the teachings of the cited

references. Booth's teaching of setting the pivot date to a predetermined value would enable users of

the Shaughnessy- Hazama's system to have direct control over the return of reformatted dates to

thereby preset the date processing system in accordance with their needs. 11. Claims 11-15 are

rejected under 35 U. S. C. 103 (a) as being unpatentable over Shaughnessy in view of Hazama, and

further in view of Booth.

As to claim 11, Shaughnessy substantially discloses the invention including the claimed

method of processing dates in a database' as a method and system for modifying and operating a

computer system to perform operations on date fields having a two digit representation for the year

without mistaking the year 2000 and the year 1900 (col.1, lines 714 et seq). Shaughnessy also

suggests the conversion of all dates within the database from a two digit format to a four digit format

as a viable, but costly alternative for the year 2000 problem (col. 1, lines 31-46 et seq). The method

as recited in the body of the claim is particularly taught by the cited references as follows:.

Shaughnessy discloses the claimed step of 'providing a database with dates stored therein according

to a format wherein M.sub.I M.sub.2 is the numerical month designator, D.sub. ID.sub.2 is the

numerical day designator, and Y.sub.I Y.sub.2 is the numerical year designator, all of dates falling

within a 10-decade period of time which includes the decade beginning in the year 2000' by providing

a database having a 6 digit-field for storing a Date type in the form of MMDDYY(see appendix in col.

18, Date Type A), wherein the MM represents the month, DD represents the day and YY represents

the year for a particular six digit-date, and wherein the ten decade includes a decade date in the 21st

century (col. 6, lines 28-29 et seq)_ As pointed out in column 2, lines 11-14 and column 3, lines 4-8 of

Patent No. 5,806,063, all dates in commercial and industrial databases span within one 100 year.

Shaughnessy's system being of the commercial or industrial kind described in the cited patent, must

therefore, as a practical matter, inherently incorporate this limitation. Shaughnessy also discloses a

subroutine for determining the current date2 to thereby select a 100 year cycle wherein the current

date is the pivot date and wherein the cycle ends a 100 years from said current date (col.5, lines

31-36). Additionally, Shaughnessy discloses the step of 'determining a century designator

C. sub. I C.sub.2 for each date in the database, C. sub. I C.sub.2 having a first value if Y.sub.I

Y.sub.2 is less than Y. sub. A Y. sub. B and having a second value if Y. sub. I Y.sub.2 is equal

to or greater than Y. sub. A Y. sub. B' as the comparison of the current date to the date when

the system was installed with the modifications (modified system install date) to thereby The current

date, by virtue of being the pivot date in the 100 year-cycle and by being initially set to the operating

system date, is therefore not any later than the earliest date in 100 year-cycle in the determine the

century value (co1.5, lines 36-65 et seq). Alternatively, Shaughnessy discloses the comparison of the

YYMMDD portion 3 of the date to the corresponding date portion at the end of the 100 year cycle to

thereby determine the century value (co1.7, lines 7-15 et seq). Finally, Shaughnessy discloses the

step of reformatting each date in the form C. sub. I C.sub.2 Y.sub. I Y.sub.2 M.sub. I M.sub.2 D.sub. I

D.sub.2 to facilitate further processing of the dates' by appending the determined century value

before the YYMMDD date in order to yield a CCYYMMDD date format (col.5, lines 46-51; co1.6, lines

57-65 et seq), and by returning one date field with the converted date to the subroutine and a means

for returning a parameter to the application program for use in further operations (col. 1, lines 47-54 et

seq).

It is noted, however, Shaughnessy does not particularly detail that the YaYb value for the first

decade of the window is no later than the earliest YIY2 year designator in the database in the

selection of the 10 decade window. On the other hand, Hazama discloses an analogous date

processing system having two digit dates spanning from the 20th century to the 21st century stored in

memory (page 2, claim 1) wherein, for a 100 year window (restricting the all the dates in the database

between the 20th and 21st centuries), the pivot date for the window is selected based on a two digit

year date that is smaller than the smallest two digit year date in the database (e.g. if smallest two digit

date stored in database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of

translated document, last paragraph). It would have been obvious to one of ordinary skill in the art of

data processing to combine the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date,

which compares low to all other dates (col. 7, lines 16-17 et seq).teachings of the cited references.

The ordinary skilled artisan having read Shaughnessy would immediately see the need to determine

which 100 year span to use. This determination would have led the ordinary skilled artisan to the

Hazama reference,, which teaches the pivot date being smaller than the smallest two digit date in the

database having all the dates within a 100 year period as a solution to restrict the selection of

Shaughnessy's window and thereby forcing all dates already stored in the database to fall in the 20th

century.

Shaughnessy and Hazama do not specifically disclose the step of 'sorting the dates in the form

$C_1 C_2 Y_1 Y_2 M_1 M_2 D_1 D_2$.' However, Booth discloses an

analogous system that utilizes the Clipper programming language to process dates stored in a

database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq. In particular, analogously

to Shaughnessy, Booth discloses the SET EPOCH command for comparing an entered two digit date

with the year digit (pivot date) of the epoch setting to determine the century to place the date into (p.

941, see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a

corresponding eight digit date (C1C2Y1Y2M1M2D1D2) (see p. 940-941). Additionally, Booth

complements Shaughnessy by suggesting the sorting of converted dates after having been

reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary

skill in the art of data processing at the time of the instant invention to combine the teachings of the

cited references. Booth's teaching of sorted reformatted dates would facilitate Shaughnessy Hazama'

s system to return the reformatted dates in chronological sequence. . Such disclosed sorting would

therefore be very useful when indexing the database in date order, as suggested by Booth on page

945.

As to claim 12, Shaughnessy, Hazama and Booth disclose the invention as discussed in the

rejection of claim 11. Additionally, Shaughnessy discloses step of converting pre-existing date

information having a different format into the format wherein $M_1 M_2$ is the numerical month

designator, $D_1 D_2$ is the numerical day designator and $Y_1 Y_2$ is the numerical year

designator' by converting the current date in a six digit format (YYMMDD), wherein Y'Y' represents

the year, MM represents the month and DD represents the day (col. 8, lines 18-27 et seq).

As to claim 13, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection

of claim 11. Booth further complements Shaughnessy by disclosing the claimed step of 'selecting Y.

sub. A Y. sub. B such that Y. sub. B is 0 (zero) ' by suggesting that the pivot date be set to 90 and by

selecting set epoch to be 1990 (i.e. YAYB = 90), such that YB equals to zero (p. 942). It would have

been obvious to one of ordinary skill in the art of data processing at the time of the instant invention to

combine the teachings of the cited references. Booth's teaching of setting the pivot date to a

predetermined value would enable users of the Shaughnessy- Hazama' s system to have direct

control over the return of reformatted dates to thereby preset the date processing system in

accordance with their needs.

As to claim 14, Shaughnessy, Hazama and Booth disclose the invention as discussed in the

rejection of claim 11. Booth further complements Shaughnessy by disclosing the step of 'storing the

sorted dates and their associated information back into the database' by renaming and storing sorted

dates in the CUSTIVIER.DBF NEW CUST.DBF databases (p. 841). It would have been obvious to

one of ordinary skill in the art of data processing to combine the teachings of the cited references.

Booth's teaching of the storing the sorted reformatted dates would allow users of

Shaughnessy-Hazama's system to readily retrieve the reformatted dates in their chronological

sequence at any time and in accordance with their needs.

As to claim 15, Shaughnessy, Hazama and Booth disclose the invention as discussed in the

rejection of claim 14. Additionally, Booth discloses the step of 'manipulating information in the

database having the reformatted date therein' by renaming and storing the sorted dates in the

CUSTMER.DBF NEW CUST.DBF databases (p. 841). It would have been obvious to one of ordinary

skill in the art of data processing to combine the teachings of the cited references. Booth's teaching of

the storing the sorted reformatted dates would users of Shaughnessy-Hazama's system to readily

retrieve the reformatted dates in their chronological sequence at any time and in accordance with

their needs.

12. Claims 16-18, 20, 22, 24-25 are rejected under 35 U.S.C. 103(a) as obvious over Shaughnessy in

view Hazama.

As to claim 16, Shaughnessy substantially discloses the invention including the claimed

method of processing symbolic representations of dates stored in a database' as a method

and system for modifying and operating a computer system to perform operations on date

fields having a two digit representation for the year without mistaking the year 2000 and the

year 1900 (col. 1, lines 7-14 et seq). Shaughnessy also suggests the conversion of all dates

within the database from a two digit format to a four digit format as a viable, but costly

alternative for the year 2000 problem (col. 1, lines 31-46 et seq).

The method as recited in the body of the claim is particularly taught by the cited references

as follows:

Shaughnessy discloses the claimed 'method of processing symbolic representations of dates

stored in a database' as a method and system for modifying and operating a computer system to

perform operations on date fields having a two digit representation for the year without mistaking the

year 2000 and the year 1900 (col. 1, lines 7-14 et seq). In particular, Shaughnessy discloses the step

of 'providing a database with symbolic representations of dates stored therein according to a format

wherein $M_1 M_2$ is the numerical month designator, $D_1 D_2$ is the numerical day

Application/Control Number⬤9/512,592; 90/005,592; 90/005,62⬤0/005727;   Page
90/006,541   24
Art Unit: 2161

designator, and Y_1Y_2 is the numerical year designator' as a database having a 6 digit-field

for storing a Date type in the form of MMDDYY(see appendix in col. 18, Date Type A), wherein the

MM represents the month, DD represents the day and YY represents the year for a particular six

digit-date. Further, Shaughnessy discloses the claimed 'all of the symbolic representations of dates

falling within a 10-decade period of time' as a date having a cycle or a range of a 100 years

(col. 18, Cycle/Range CI = THE DATE CYCLE IS 100 YEARS). As pointed out in column

2, lines 11-14 and column 3, lines 4-8 of Patent No. 5,806,063, all dates in commercial and

industrial databases span within one 100 year. Shaughnessy's system being of the

commercial or industrial kind described in the cited patent, must therefore, as a practical

matter, inherently incorporate this limitation. Shaughnessy also discloses a subroutine for

determining the current dates to thereby select a 100 year cycle wherein the current date is

the pivot date and wherein the cycle ends a 100 years from said current date (col.5, lines 31

36). Additionally, Shaughnessy discloses the step of determining a century designator

C_1C_2 for each symbolic representation of a date in the database, C_1C_2

having a first value if Y_1Y_2 is less than Y_AY_B and having a second

value if Y_1Y_2 is equal to or greater than Y_AY_B' as the comparison of

the current date to the date when the system was installed with the modifications (modified

system install date) to thereby determine the century value (col.5, lines 36-65 et seq).

Alternatively, Shaughnessy discloses the comparison of the YYMMDD portion' of the date

to the corresponding date portion at the end of the 100 year cycle to thereby determine the

century value (col.7, lines 7-15 et seq). Finally, Shaughnessy discloses the step of '

reformatting the symbolic representation of each symbolic representation of a date in the

database, without the addition of any new data field to the database, with the reformatted

symbolic representation of each date in the database having the values CI C2, Y1 Y2, MI

M2, and DI D2, in order to facilitate collectively further processing the reformatted symbolic

4The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set

to the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any

later than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date

which compares low to all other dates (col. 7, lines 16-17 et seq). representations of each of the

symbolic representations of each of the dates.' by appending the determined century value before the

YYMMDD date in order to yield a CCYYMMDD date format, whereby said appending was performed

without adding a new field to the database (col.5, lines 46-51; col.6, lines 57-65 et seq), and by

returning one date field with the converted date to the subroutine and a means for returning a

parameter to the application program for use in further operations (col. 1, lines 47-54 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the

window is no later than the earliest YIY2 year designator in the database in the selection of

the 10 decade window. However, Hazama discloses an analogous date processing system

having two digit dates spanning from the 20th century to the 21st century stored in memory

(page 2, claim 1) wherein, for a 100 year window (restricting the all the dates in the database

between the 20th and 21st centuries), the pivot date for the window is selected based on a

two digit year date that is smaller than the smallest two digit year date in the database (e.g. if smallest

two digit date stored in database is 73, the pivot date for the 100 year window is chosen to be 72).

(page 4 of translated document, last paragraph). It would have been obvious to one of ordinary skill in

the art of data processing to combine the teachings of the cited references. The ordinary skilled

artisan having read Shaughnessy would immediately see the need to determine which 100 year span

to use. This determination would have led the ordinary skilled artisan to the Hazama reference, which

teaches the pivot date being smaller than the smallest two digit date in the database having all the

dates within a 100 year period as a solution to restrict the selection of Shaughnessy's window and

thereby forcing all dates already stored in the database to fall in the 20th century.

As to claim 17, Shaughnessy and Hazama disclose the invention as discussed in the

rejection of claim 16. Further, Shaughnessy discloses the claimed limitation whereby 'the

window includes at least a portion of the decade beginning in the year 2000' by suggesting

the use of a 100 year window that includes a decade date in the 21st century (col. 6, lines 28

29 et seq).

As to claim 18, Shaughnessy and Hazama disclose the invention as discussed in the

rejection of claim 17. Further, Shaughnessy discloses the claimed limitation whereby, 'the

step of determining includes the step of determining the first value as 20 and the second

value as 19' by assigning the century value to 19 if the YYDDD portion of the date is greater than or

equal to the corresponding portion of the corresponding portion of the modified system install date

(col. 5, lines 40-46) and by assigning the century value to 20 if the pivot date is less than the modified

system install date (col.5, lines 52-60 et seq).

As to claim 19, Shaughnessy and Hazama substantially disclose the invention as

discussed in the rejection of claim 16 above. Shaughnessy and Hazama do not particularly

disclose the additional step of 'sorting the symbolic representations of dates, after the step of

reformatting.' However, Booth discloses an analogous system that utilizes the Clipper

programming language to process dates stored in a database to thereby derive other dates

therefrom (p. 939, lines 1-3 et seq). In particular, analogously to Shaughnessy, Booth

discloses the SET EPOCH command for comparing an entered two digit date with the year

digit (pivot date) of the epoch setting to determine the century to place the date into (p. 941,

see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a

corresponding eight digit date (C 1 C2Y 1 Y2M I M2D 1 D2) (see p. 940-941). Additionally,

Booth complements Shaughnessy by suggesting the sorting of converted dates after having

been reformatted by the SET EPOCH command (p. 945). It would have been obvious to one

of ordinary skill in the art of data processing at the time of the instant invention to combine

the teachings of the cited references. Booth's teaching of sorted reformatted dates would

facilitate Shaughnessy-Hazama's system to return the reformatted dates in chronological

sequence. . Such disclosed sorting would therefore be very useful when indexing the

database in date order, as suggested by Booth on page 945.

As to claim 20, Shaughnessy and Hazama disclose the invention as discussed in the rejection

of claim 16,. Further, Shaughnessy discloses the claimed limitation, wherein 'the step of reformatting

includes the step of reformatting each symbolic representation of a date into the format C_1

C_2 Y_1 Y_2 M_1 M_2 D_1 D_2 separately from the symbolic

representations in the database' as the conversion of the current date from a six digit format

(YYMMDD) into an 8-digit format (CCYYMMDD) (col. 5, lines 48-50 et seq).

As to claim 22, Shaughnessy and Hazama disclose the invention as discussed in the

rejection of claim 16. Further, Shaughnessy discloses the claimed limitation, wherein the step of providing a database includes the step of converting pre-existing date information having a different format into the format wherein $M.sub. I M.sub.2$ is the numerical month designator, $D. sub. I D.sub.2$ is the numerical day designator and $Y. sub. I Y.sub.2$ is the numerical year designator' by as the converting the current date in a six digit format (YYMMDD), wherein YY represents the year, MM represents the month and DD represents the day (col. 8, lines 18-27 et seq).

As to claim 24, Shaughnessy and Hazama disclose the invention as discussed in the rejection of claim 16,. Further, Shaughnessy discloses the claimed limitation of `storing the symbolic representation of dates and their associated information back into the database after the step of reformatting' by saving the converted date in the database (col. 6, lines 1-3 et seq).

As to claim 25, Shaughnessy and Hazama disclose the invention as discussed in the rejection of claim 24. Further, Shaughnessy discloses the claimed limitation of' manipulating information in the database having the reformatted date information therein' by performing updates on the converted dates and saving said converted dates in the database (col. 6, lines 1-22 et seq.)

13. Claims 19, 21, 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view of Hazama, as applied to the rejection of claims 16-18, 20, 22, 24-25 above, further in view of Booth.

As to claim 21, Shaughnessy and Hazama substantially disclose the invention as

discussed in the rejection of claim 20 above. Shaughnessy and Hazama do not specifically

disclose the additional step of 'sorting the symbolic representations of dates using a

numerical-order sort, after the step of reformatting.' However, Booth discloses an analogous

system that utilizes the Clipper programming language to process dates stored in a database

to thereby derive other dates therefrom (p. 939, lines 1-3 et seq. In particular, analogously to

Shaughnessy, Booth discloses the SET EPOCH command for comparing an entered two digit

date with the year digit (pivot date) of the epoch setting to determine the century to place the

date into (p. 941, see SET EPOCH paragraph et seq) thereby converting a six digit date

(MMDDYY) into a corresponding eight digit date (C 1 C2YI Y2M 1 M21) 1 D2) (see p. 940

941). Additionally, Booth complements Shaughnessy and Hazama by suggesting the sorting

of converted dates after having been reformatted by the SET EPOCH command (p. 945). It

would have been obvious to one of ordinary skill in the art of data processing at the time of

the instant invention to combine the teachings of the cited references. Booth's teaching of

sorted reformatted dates would facilitate the Shaughnessy-Hazama' s system to return the

reformatted dates in chronological sequence. . Such disclosed sorting would therefore be

very useful when indexing the database in date order, as suggested by Booth on page 945.

As to claim 23, Shaughnessy and Hazama substantially disclose the invention as

discussed in the rejection of claim 16. Shaughnessy and Hazama do not specifically,

disclose the step of selecting Y.sub.A Y.sub.B such that Y.sub.B is 0 (zero). However,

Booth discloses an analogous system that utilizes the Clipper programming language to

process dates stored in a database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq.

In particular, analogously to Shaughnessy, Booth discloses the SET EPOCH command for comparing

an entered two digit date with the year digit (pivot date) of the epoch setting to determine the century

to place the date into (p. 941, see SET EPOCH paragraph et seq) thereby converting a six digit date

(MMDDYY) into a corresponding eight digit date (CIC2YIY2MIM2DID2) (seep. 940-941). Additionally,

Booth complements Shaughnessy and Hazama by suggesting that the pivot date be set to 90 by

selecting set epoch to be 1990 (i.e. YAYB = 90), such that YB equals to zero (p. 942). It would have

been obvious to one of ordinary skill in the art of data processing at the time of the instant invention to

combine the teachings of the cited references. Booth's teaching of setting the pivot date to a

predetermined value would enable users of Shaughnessy-Hazama's system to have direct control

over the return of reformatted dates to thereby preset the date processing system in accordance with

their needs.

14. Claims 26-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in

view of Hazama, and further in view of Booth.

As to claim 26, Shaughnessy substantially discloses the invention including the claimed

method of processing dates in a database' as a method and system for modifying and operating a

computer system to perform operations on date fields having a two digit representation for the year

without mistaking the year 2000 and the year 1900 (col.1, lines 714 et seq). Shaughnessy also

suggests the conversion of all dates within the database from a two digit format to a four digit format

as a viable, but costly alternative for the year 2000 problem (col. 1, lines 31-46 et seq).

The method as recited in the body of the claim is particularly taught by the cited references

as follows:

Shaughnessy discloses the claimed 'method of processing dates in a database' as a method

and system for modifying and operating a computer system to perform operations on date

fields having a two digit representation for the year without mistaking the year 2000 and the

year 1900 (cot. 1, lines 7-14 et seq). Also, Shaughnessy discloses the step of 'providing a

database with symbolic representations of dates stored therein according to a format wherein

$M_{sub.1} M_{sub.2}$ is the numerical month designator, $D_{sub.1} D_{sub.2}$ is the numerical day

designator, and $Y_{sub.1} Y_{sub.2}$ is the numerical year designator' as a database having a 6

digit-field for storing a Date type in the form of MMDDYY(see appendix in col. 18, Date

Type A), wherein the MM represents the month, DD represents the day and YY represents

the year for a particular six digit-date. Further, Shaughnessy discloses the claimed 'all of the

symbolic representations of dates falling within a 10-decade period of time' as a date having

a cycle or a range of a 100 years (col. 18, Cycle/Range C 1 = THE DATE CYCLE IS 100

YEARS). As pointed out in column 2, lines 11-14 and column 3, lines 4-8 of Patent No.

5,806,063, all dates in commercial and industrial databases span within one 100 year.

Shaughnessy also discloses a subroutine for determining the current date 6 to thereby select a

100 year cycle wherein the current date is the pivot date and wherein the cycle ends a 100

6The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set

to the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any

later years from said current date (col.5, lines 31-36). Additionally, Shaughnessy discloses the step of

determining a century designator $C_{sub.1} C_{sub.2}$ for each symbolic representation of a date in the

database, $C_{sub.1} C_{sub.2}$ having a first value if $Y_{sub.1} Y_{sub.2}$ is less than $Y_{sub.A} Y_{sub.B}$ and

having a second value if $Y_{sub.1} Y_{sub.2}$ is equal to or greater than $Y_{sub.A} Y_{sub.B}$' as the

comparison of the current date to the date when the system was installed with the modifications

(modified system install date) to thereby determine the.century value (col.5, lines 36-65 et seq).

Alternatively, Shaughnessy discloses the comparison of the YYMMDD portion' of the date to the

corresponding date portion at the end of the 100 year cycle to thereby determine the century value

(col.7, lines 7-15 et seq). Finally, Shaughnessy discloses the step of ` reformatting the symbolic

representation of each symbolic representation of a date in the database, without the addition of any

new data field to the database, with the reformatted symbolic representation of each date in the

database having the values C 1 C2, Y 1 Y2, MI M2, and DI D2, in order to facilitate collectively further

processing the reformatted symbolic representations of each of the symbolic representations of each

of the dates.' by appending the determined century value before the YYMNIDD date in order to yield a

CCYYMMDD date format, whereby said appending was performed without adding a new field to the

database (col.5, lines 46-51; col.6, lines 57-65 et seq), and by returning one date field with the

converted date to the subroutine and a means for returning a parameter to the application program

for use in further operations (col. 1, lines 47-54 et seq). than the earliest date in 100 year-cycle in the

database. Shaughnessy specifically suggests that it might be desirable to set the current date to a

date, which compares low to all other dates (col. 7, lines 16-17 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest YI Y2 year designator in the database in the selection of the 10 decade window.

However, Hazama discloses an analogous date processing system having two digit dates spanning

from the 20th century to the 21st century stored in memory (page 2, claim 1) wherein, for a 100 year

window (restricting the all the dates in the database between the 20th and 21st centuries), the pivot

date for the window is selected based on a two digit year date that is smaller than the smallest two

digit year date in the database (e.g. if smallest two digit date stored in database is 73, the pivot date

for the 100 year window is chosen to be 72). (page 4 of translated document, last paragraph). It

would have been obvious to one of ordinary skill in the art of data processing to combine the

teachings of the cited references. The ordinary skilled artisan having read Shaughnessy would

immediately see the need to determine which 100 year span to use- This determination would have

led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being smaller

than the smallest two digit date in the database having all the dates within a 100 year period as a

solution to restrict the selection of Shaughnessy's window and thereby forcing all dates already stored

in the database to fall in the 20th century.

Shaughnessy and Hazama do not specifically disclose the step of `sorting the dates in the

form $C.sub.1$ $C.sub.2$ $Y.sub.1$ $Y.sub.2$ $M.sub.1$ $M.sub.2$ $D.sub.1$ $D.sub.2$.' However, Booth discloses

an analogous system that utilizes the Clipper programming language to process dates stored in a

database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq. In particular, analogously

to Shaughnessy, Booth discloses the SET EPOCH command for comparing an entered two digit date

with the year digit (pivot date) of the epoch setting to determine the century to place the date into (p.

941, see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a

corresponding eight digit date (C1C2Y1Y2M1M2D1D2) (see p. 940-941). Additionally, Booth

complements Shaughnessy and Hazama by suggesting the sorting of converted dates after having

been reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of

ordinary skill in the art of data processing at the time of the instant invention to combine the teachings

of the cited references. Booth's teaching of sorted reformatted dates would facilitate the

Shaughnessy-Hazama's system to return the reformatted dates in chronological sequence. .

Such disclosed sorting would therefore be very useful when indexing the database in date

order, as suggested by Booth on page 945.

As to claim 27, Shaughnessy, Hazama and Booth disclose the invention as discussed in

the rejection of claim 26. Additionally, Shaughnessy discloses step of 'converting

pre-existing date information having a different format into the format wherein M.sub. I

M.sub.2 is the numerical month designator, D.sub. I D.sub.2 is the numerical day designator

and Y.sub. I Ysub.2 is the numerical year designator' by converting the current date in a six

digit format (YYMMDD), wherein YY represents the year, MM represents the month and

DD represents the day (col. 8, lines 18-27 et seq).

As to claim 28, Shaughnessy, Hazama and Booth disclose the invention as discussed in the

rejection of claim 26. Booth further complements Shaughnessy and Hazama by disclosing the

claimed step of 'selecting Y.sub.A Y.sub.B such that Y.Sub.B is 0 (zero) ' by suggesting that the pivot

date be set to 90 and by selecting set epoch to be 1990 (i.e. YAYB = 90), such that YB equals to zero

(p. 942). It would have been obvious to one of ordinary skill in the art of data processing at the time of

the instant invention to combine the teachings of the cited references. Booth's teaching of setting the

pivot date to a predetermined value would enable users of Shaughnessy-Hazama's system to have

direct control over the return of reformatted dates to thereby preset the date processing system in

accordance with their needs. As to claim 29, Shaughnessy, Hazama and Booth disclose the invention

as discussed in the rejection of claim 26. Booth further complements Shaughnessy by disclosing the

step of 'storing the sorted dates and their associated information back into the database' by renaming

and storing sorted dates in the CUSTMER.DBF NEW CUST.DBF databases (p. 841). It would have

been obvious to one of ordinary skill in the art of data processing to combine the teachings of the

cited references. Booth's teaching of the storing the sorted reformatted dates would users of

Shaughnessy-Hazama's system to readily retrieve the reformatted dates in their chronological

sequence at any time and in accordance with their needs.

As to claim 30, Shaughnessy, Hazama and Booth disclose the invention as discussed in the

rejection of claim 29. Additionally, Booth discloses the step of 'manipulating information in the

database having the reformatted date therein' by renaming and storing the sorted dates in the

CUSTMER.DBF NEW CUST.DBF databases (p. 841). It would have been obvious to one of ordinary

skill in the art of data processing to combine the teachings of the cited references. Booth's teaching of

the storing the sorted reformatted dates would users of Shaughnessy- Hazama's system to readily

retrieve the reformatted dates in their chronological sequence at any time and in accordance with

their needs.

15. Claim 31 is rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view of

Hazama.

As to claim 31, Shaughnessy substantially discloses the invention including the claimed

'method of processing symbolic representations of dates stored in a database' as a method and

system for modifying and operating a computer system to perform operations on date fields having a

two digit representation for the year without mistaking the year 2000 and the year 1900 (col.1, lines

7-14 et seq). Shaughnessy also suggests the conversion of all dates within the database from a two

digit format to a four digit format as a viable, but costly alternative for the year 2000 problem (col. 1,

lines 31-46 et seq). The method as recited in the body of the claim is particularly taught by the cited

references as follows:

Shaughnessy discloses the step of providing a database with symbolic representations of dates

stored therein according to a format wherein $Y.sub.1 Y.sub.2$ is the numerical year designator' as a

database having a 6 digit-field for storing a Date type in the form of MMDDYY(see appendix in col.

18, Date Type A), wherein the MM represents the month, DD represents the day and YY represents

the year for a particular six digit-date having a cycle or a range of a 100 years (col. 18, Cycle/Range

C I = THE DATE CYCLE IS 100 YEARS). As pointed out in column 2, lines 11-14 and column 3, lines

4-8 of Patent No. 5,806,063, all dates in commercial and industrial databases span within one 100

year.

Shaughnessy's system being of the commercial or industrial kind described in the cited

patent, must therefore, as a practical matter, inherently incorporate this limitation.

Shaughnessy also discloses a subroutine for determining the current dates to thereby select a

100 year cycle wherein the current date is the pivot date and wherein the cycle ends a 100

years from said current date (col.5, lines 31-36). Additionally, Shaughnessy discloses the

step of 'determining a century designator $C.sub.1 C.sub.2$ for each symbolic representation of a date

in the database, $C.sub.1 C.sub.2$ having a first value if $Y.sub.1 Y.sub.2$ is less than $Y.sub.A Y.sub.B$

and having a second value if $Y.sub.1 Y.sub.2$ is equal to or greater than $Y.sub.A Y.sub.B$' as the

comparison of the current date to the date when the system was installed with the modifications

(modified system install date) to thereby determine the century value (col.5, lines 36-65 et seq).

Alternatively, Shaughnessy discloses the comparison of the YYMMDD portion9 of the date to the

corresponding date portion at the end of the 100 year cycle to thereby determine the century value

(col.7, lines 7-15 et seq). Finally, Shaughnessy discloses the step of 'reformatting the symbolic

representation of each symbolic representation of a date in the database, without the addition of any

new data field 'The current date, by virtue of being the pivot date in the 100 year-cycle and by being

initially set to the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is

therefore not any later than the earliest date in 100 year-cycle in the database. Shaughnessy

specifically suggests that it might be desirable to set the current date to a date, which compares low

to all other dates (col. 7, lines 16-17 et seq). to the database, with the reformatted symbolic

representation of each date in the database having the values C1 C2, Y1 Y2, MI M2, and DI D2, in

order to facilitate collectively further processing the reformatted symbolic representations of each of

the symbolic representations of each of the dates.' by appending the determined century value before

the YYMMDD date in order to yield a CCYYMMDD date format, whereby said appending was

performed without adding a new field to the database (co1.5, lines 46-51; col.6, lines 57-65 et seq),

and by returning one date field with the converted date to the subroutine and a means for returning a

parameter to the application program for use in further operations (col. 1, lines 47-54 et seq). .

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest YIY2 year designator in the database in the selection of the 10 decade window.

However, Hazama discloses an analogous date processing system having two digit dates spanning

from the 20th century to the 21st century stored in memory (page 2, claim 1) wherein, for a 100 year

window (restricting the all the dates in the database between the 20th and 21st centuries), the pivot

date for the window is selected based on a two digit year date that is smaller than the smallest two

digit year date in the database (e.g. if smallest two digit date stored in database is 73, the pivot date

for the 100 year window is chosen to be 72). (page 4 of translated document, last paragraph). It

would have been obvious to one of ordinary skill in the art of data processing to combine the

teachings of the cited references. The ordinary skilled artisan having read Shaughnessy would

immediately see the need to determine which 100 year span to use. This determination would have

led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being smaller

than the smallest two digit date in the database having all the dates within a 100 year period as a

solution to restrict the selection of Shaughnessy's window and thereby forcing all dates already stored

in the database to fall in the 20th century.

16. Claim 32 is rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view of

Hazama, and further in view of Booth.

As to claim 32, Shaughnessy substantially discloses the invention including the claimed

`method of processing dates in a database' as a method and system for modifying and operating a

computer system to perform operations on date fields having a two digit representation for the year

without mistaking the year 2000 and the year 1900 (col.1, lines 714 et seq). Shaughnessy also

suggests the conversion of all dates within the database from a two digit format to a four digit format

as a viable, but costly alternative for the year 2000 ;problem (col. 1, lines 31-46 et seq). The method

as recited in the body of the claim is particularly taught by the cited references as follows:

Shaughnessy discloses the step of 'providing a database with symbolic representations of dates

stored therein according to a format wherein Y.sub.1 Y.sub.2 is the numerical year designator' as a

database having a 6 digit-field for storing a Date type in the form of MMDDYY(see appendix in col.

18, Date Type A), wherein the MM represents the month, DD represents the day and YY represents

the year for a particular six digit-date falling within a 10-decade period of time' as a date having a

cycle or a range of a 100 years (col. 18, Cycle/Range CI = THE DATE CYCLE IS 100 YEARS). As

pointed out in column 2, lines 11-14 and column 3, lines 4-8 of Patent No. 5,806,063, all dates in

commercial and industrial databases span within one 100 year. Shaughnessy's system being of the

commercial or industrial kind described in the cited patent, must therefore, as a practical

matter, inherently incorporate this limitation. Shaughnessy also discloses a subroutine for

determining the current date to thereby select a 100 year cycle wherein the current date is

the pivot date and wherein the cycle ends a 100 years from said current date (col.5, lines 31

36). Additionally, Shaughnessy discloses the step of determining a century designator

$C_1 C_2$ for each symbolic representation of a elate in the database, $C_1 C_2$

having a first value if $Y_1 Y_2$ is less than $Y_A Y_B$ and having a second

value if $Y_1 Y_2$ is equal to or greater than $Y_A Y_B$' as the comparison of

the current date to the date when the system was installed with the modifications (modified

system install date) to thereby determine the century value (col_5, lines 36-65 et seq).

Alternatively, Shaughnessy discloses the comparison of the YYMMDD portion" of the date

to the corresponding date portion at the end of the 100 year cycle to thereby determine the

century value (col.7, lines 7-15 et seq). Finally, Shaughnessy discloses the step of

'reformatting the symbolic representation of each of the dates in the database, without the

addition of any new data field to the database, with the reformatted symbolic representation

of each date in the database having the values C1 C2, Y1 Y2, MI M2, and DI D2, in order to

'°The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set

to the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any

later than the earliest date in 100 year-cycle in the database.

" Shaughnessy specifically suggests that it might be desirable to set the current date to a date

which compares low to all other dates (col. 7, lines 16-17 et seq).

facilitate collectively further processing the reformatted symbolic representations of each of the

symbolic representations of each of the dates' by appending the determined century value before the

YYMMDD date in order to yield a CCYYIVLMDD date format, whereby said appending was

performed without adding a new field to the database (col.5, lines 46-51; co1.6, lines 57-65 et seq),

and by returning one date field with the converted date to the subroutine and a means for returning a

parameter to the application program for use in further operations (col. 1, lines 47-54 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest YlY2 year designator in the database in the selection of the I 0 -decade window.

However, Hazama discloses an analogous date processing system having two digit dates spanning

from the 20th century to the 21st century stored in memory (page 2, claim 1) wherein, for a 100 year

window (restricting the all the dates in the database between the 20th and 21st centuries), the pivot

date four the window is selected based on a two digit year date that is smaller than the smallest two

digit year date in the database (e.g. if smallest two digit date stored in database is 73, the pivot date

for the 100 year window is chosen to be 72). (page 4 of translated document, last paragraph). It

would have been obvious to one of ordinary skill in the art of data processing to combine the

teachings of the cited references. The ordinary skilled artisan having read Shaughnessy would

immediately see the need to determine which 100 year span to use. This determination would have

led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being smaller

than the smallest two digit date in the database having all the dates within a 100 year period as a

solution to restrict the selection of Shaughnessy's window and thereby forcing all dates

already stored in the database to fall in the 20th century.

Shaughnessy and Hazama do not specifically, disclose the step of' sorting the dates in

the form $C.sub. I C.sub.2 Y.sub. I Y.sub.2 M.sub.I M.sub.2 D.sub. I D.sub.2.'$ However,

Booth discloses an analogous system that utilizes the Clipper programming language to

process dates stored in a database to thereby derive other dates therefrom (p. 939, lines 1-3 et

seq. In particular, analogously to Shaughnessy, Booth discloses the SET EPOCH command

for comparing an entered two digit date with the year digit (pivot date) of the epoch setting to

determine the century to place the date into (p. 941, see SET EPOCH paragraph et seq) thereby

converting a six digit date (MMDDYY) into a corresponding eight digit date (C 1 C2Y 1 Y2M 1 M2D 1

D2) (see p. 940-941). Additionally, Booth complements Shaughnessy by suggesting the sorting of

converted dates after having been reformatted by the SET EPOCH command (p. 945). It would have

been obvious to one of ordinary skill in the art of data processing at the time of the instant invention to

combine the teachings of the cited references. Booth's teaching of sorted reformatted dates would

facilitate the Shaughnessy Hazama's system to return the reformatted dates in chronological

sequence. . Such disclosed sorting would therefore be very useful when indexing the database in

date order, as suggested by Booth on page 945.

17. Claim 33 is rejected under 35 L.S.C. 103(a) as being unpatentable over Shaughnessy in view of

Hazama.

As to claim 33, Shaughnessy substantially discloses the invention including the claimed

method of processing symbolic representations of dates stored in a database' as a method

and system for modifying and operating a computer system to perform operations on date

fields having a two digit representation for the year without mistaking the year 2000 and the

year 1900 (col. 1, lines 7-14 et seq). Shaughnessy also suggests the conversion of all dates

within the database from a two digit format to a four digit format as a viable, but costly

alternative for the year 2000 problem (col. 1, lines 31-46 et. seq). The method as recited in

the body of the claim is particularly taught by the cited references as follows:

Shaughnessy discloses the step of 'providing a database with symbolic representations of dates

stored therein according to a format wherein Y1 Y2 is the numerical year designator' as a database

having a 6 digit-field for storing a Date type in the form of MMDDYY(see appendix in col. 18, Date

Type A), wherein the MM represents the month, DD represents the day and YY represents the year

for a particular six digit-date having a cycle or a range of a 100 years (col. 18, Cycle/Range CI ==

THE DATE CYCLE IS 100 YEARS). As pointed out in column 2, lines 11-14 and column 3, lines 4-8

of Patent No. 5,806,063, all dates in commercial and industrial databases span within one 100 year.

Shaughnessy's system being of the commercial or industrial kind described in the cited patent, must

therefore, as a practical matter, inherently incorporate this limitation Shaughnessy also discloses a

subroutine for determining the current date 12 to thereby select a 100 year cycle wherein the current

date is the pivot date and wherein the cycle ends a 100 years from said current date (co1.5, lines

31-36). Additionally, Shaughnessy discloses the step of 'determining a century'

The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set to the

operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any late

designator C1 C2 for each symbolic representation of a date in the database, CI C2 having a first

value if YI Y2 is less than A YB and having a second value if YI Y2 is equal to or greater than A YB'

as the comparison of the current date to the date when the system was installed with the

modifications (modified system install date) to thereby determine the century value (col.5, lines 36-65

et seq). Alternatively., Shaughnessy discloses the comparison of the YYMMDD portion13 of the date

to the corresponding date portion at the end of the 100 year cycle to thereby determine the century

'value (col.7, lines 7-15 et seq). Finally, Shaughnessy discloses the step of 'reformatting the symbolic

representation of each symbolic representation of a date in the database without changing any of the

symbolic representations of a date in the database during the reformatting step, with the reformatted

symbolic representation of each date in the database having the values C1 C2, YIY2, in order to

facilitate collectively further processing the reformatted symbolic representations of each of the dates'

by appending the determined century value before the YYMMDD date in order to yield a

CCYYMMDD date format, whereby said appending was performed without changing any of the

symbolic representations of a date in the database (col.5, lines 46-51; col.6, lines 57-65 et seq), and

by returning one date field with the converted date to the subroutine and a means for returning a

parameter to the application program for use in further operations (col. 1, lines 47-54 et seq).

Shaughnessy does not particularly detail that, the YaYb value for the first decade of

than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date, which

compares low to all other dates (col. 7, lines 16-17 et seq).

the window is no later than the earliest. Y 1 Y2 year designator in the database in the selection of the

10-decade window. However, Hazama discloses an analogous date processing system having two

digit dates spanning from the 20th century to the 21st century stored in memory (page 2, claim 1)

wherein, for a 100 year window (restricting the all the dates in the database between the 20th and

21st centuries), the pivot date for the window is selected based on a two digit year date that is smaller

than the smallest two digit year date in the database (e.g. if smallest two digit date stored in database

is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated document, last

paragraph). It would have been obvious to one of ordinary skill in the art of data processing to

combine the teachings of the cited references. The ordinary skilled artisan having read Shaughnessy

would immediately see the need to determine which 100 year span to use. This determination would

have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being

smaller than the smallest two digit date in the database having all the dates within a 100 year period

as a solution to restrict the selection of Shaughnessy's window and thereby forcing all dates already

stored in the database to fall in the 20th century.

18. Claims 34-59 are rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in

view of Hazama, and further in view of Booth method.

As to claim 34, Shaughnessy substantially discloses the invention including the claimed

`method for representing and utilizing dates stored in at least one date field of a database

utilizing symbolic representations of the dates stored in the at least one date field of the

database, which are in a format that creates ambiguity between dates in each of a pair of

adjacent centuries' as a method and system for modifying and operating a computer system

to perform operations on date fields having a two digit representation for the year without

mistaking the year 2000 and the year 1900 (col.1, lines 7-14 et seq). Shaughnessy also

suggests the conversion of all dates within the database from a two digit format to a four digit format

as a viable, but costly alternative for the year 2000 problem (col. 1, lines 31-46 et seq). The method

as recited in the body of the claim is particularly taught by the cited references as follows:

Shaughnessy discloses the step of 'converting each of the symbolic representations of dates stored in

the at least one date field of the database; to a symbolic representation of each of the respective

dates that does not create the ambiguity, by windowing the symbolic representations of each of the

respective dates as stored in the at least one date field of the database against a pivot year

represented by one of the symbolic representations of the dates as stored in the at least one date

field of the database, without the addition of any new data field to the database for purposes of such

windowing and converting; and running a program collectively on each of the converted symbolic

representations of each of the respective dates to manipulate the dates represented by the converted

symbolic representations, separately from the date data symbolic representations contained in the at

least one date field of the database' by providing a database having a 6 digit-field for storing a Date

type in the form of MMDDYY(see appendix in col. 18, Date Type A), wherein the MM represents the

month, DD represents the day and Y'Y represents the year for a particular six digit-date

falling within a 10-decade period of time' as a date having a cycle or a range of a 100 years

(col. 18, Cycle/Range CI = THE DATE CYCLE IS 100 YEARS). Shaughnessy further

provides a subroutine for determining the current date" to thereby select a 100 year cycle

wherein the current date is the pivot date and wherein the cycle ends a 100 years from said

current date (co1.5, lines 31-36). Additionally, Shaughnessy discloses the comparison of the

current date to the date when the system w-as installed with the modifications (modified

system install date) to thereby determine the century value (col.5, lines 36-65 et seq).

Alternatively, Shaughnessy provides a subroutine that compares the YYMMDD portion' 5 of

the date to the corresponding date portion at the end of the 100 year cycle to thereby

determine the century value (col.7, lines 7-15 et seq). Finally, Shaughnessy provides a

subroutine for appending the determined century value before the YYMMDD date in order to yield a

CCYYMMDD date format, whereby said appending was performed without adding a new field to the

database (co1.5, lines 46-51; col.6, lines 57-65 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest YIY2 year designator in the database in the selection of the 10 decade window.

However, Hazama discloses an analogous date processing system having two digit dates spanning

form the 20th century to the 21st century stored in memory (page 2, claim 1) wherein, for a 100 year

window (restricting the all the dates in the database

Application/Control Number● ●/512,592; 90/005,592; 90/005,62●●0/005727;   Page
90/006,541   46
Art Unit: 2161

The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set to the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any later than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date, which compares low to all other dates (col. 7, lines 16-17 et seq).

between the 20th and 21st centuries), the pivot date for the window is selected based on a two digit year date that is smaller than the smallest two digit year date in the database (e.g. If smallest two digit date stored in database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated document, last paragraph). It would have been obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited references. The ordinary skilled artisan having read Shaughnessy would immediately see the need to determine which 100 year span to use. This determination would have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being smaller than the smallest two digit date in the database having all the dates within a 100 year period as a solution to restrict the selection of Shaughnessy's window and thereby forcing all dates already stored in the database to fall in the 20th century.

Shaughnessy and Hazama do not specifically disclose the sorting of converted dates in the form C.sub.I C.sub.2 Y.sub.I Y.sub.2 M.sub. I M.sub.2 D.sub. I D.sub.2. However, Booth discloses an analogous system that utilizes the Clipper programming language to process dates stored in a database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq. In particular, analogously to Shaughnessy, Booth discloses the SET EPOCH command for comparing an entered two digit date with the year digit (pivot date) of the epoch setting to determine the century to place the date into (p. 941, see: SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a

corresponding eight digit date (CIC2YIY2MIM2DID2) (see p. 940-941). Additionally, Booth

complements Shaughnessy by suggesting the sorting of converted dates after having been

reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary

skill in the art of data processing at the time of the instant invention to combine the teachings of the

cited references. Booth's teaching of sorted reformatted dates would facilitate the Shaughnessy

Hazama's system to return the reformatted dates in chronological sequence. Such disclosed sorting

would therefore be very useful when indexing the database in date order, as suggested by Booth on

page 945.

As to claim 35, Shaughnessy, Hazama and Booth disclose the invention as discussed in the

rejection of claim 34. Additionally. Shaughnessy discloses step of 'opening the database prior to the

step of converting' by providing a subroutine; to retrieve a six digit date from its storage location in an

existing application program (i.e. requires opening the DB, first) prior to converting said date to an

eight digit format (col. 4, lines 29-33 et seq).

As to claim 36, Shaughnessy, Hazama and Booth disclose the invention as discussed in the

rejection of claim 34. Booth further complements Shaughnessy by disclosing the step of 'collectively

sorting the converted symbolic, representations prior to the step of running the program on the

converted symbolic representations' by suggesting that a string representation can be used to sort

and index the converted dates such that the dates appear in chronological order (page 945 et seq).

As to claim 37, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection

of claim 35. Booth further complements Shaughnessy by disclosing the step of

'collectively sorting the converted symbolic representations prior to the step of running the program on

the converted symbolic representations' by suggesting that a string representation can be used to sort

and index the converted dates such that the dates appear in chronological order (page 945 et seq).

As to claim 38, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection

of claim 34. Booth further complements Shaughnessy by disclosing the step of 'collectively

manipulating the converted symbolic representations prior to the step of running the program on the

converted symbolic representations' by suggesting a string representation can be used to sort and

index the converted dates such that the dates appear in chronological order (page 945 et seq).

As to claim 39, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection

of claim 35. Booth further complements Shaughnessy by disclosing the step of 'collectively

manipulating the converted symbolic representations prior to the step of running the program on the

converted symbolic representations' by suggesting that a string representation can be used to sort

and index the converted dates such that the dates appear in chronological order (page 945 et seq).

As to claim 40, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection

of claim 34. Booth further complements Shaughnessy by disclosing the step of `collectively sorting

the converted symbolic representations according to a different data field contained in the database

from the at least one date field, prior to the step of running the program on the converted symbolic

representations' by suggesting that a string representation can be used to sort and index the

converted dates such that the dates appear in chronological order (page 945 et seq), whereby said

dates, contained ill a field of the database, are sorted accordingly in a different field of the database

(page 839-40 et seq).

Application/Control Number ●9/512,592; 90/005,592; 90/005,62● ●0/005727;   Page
90/006,541   49
Art Unit: 2161

As to claim 41, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 35. Booth further complements Shaughnessy by disclosing the step of 'collectively sorting the converted symbolic representations according to a different data field contained in the database from the at least one date field, prior to the step of running the program on the converted symbolic representations' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 42, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 34. Booth further complements Shaughnessy by disclosing the step of 'collectively manipulating the converted symbolic representations according to a different data field contained in the database from the at least one date field, prior to the step of running the program on the converted symbolic representations' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 43, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 35. Booth further complements Shaughnessy by disclosing the step of 'collectively manipulating the converted symbolic representations according to a different data entry field contained in the database from the at least one date field, prior to the step of running the program on the converted symbolic representations' by suggesting that a string

representation can be used to sort and index the converted dates such that the dates appear in

chronological order (page 945 et seq), whereby said dates, contained in a field of the

database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 44, Shaughnessy, Hazama and Booth disclose the invention as discussed in

the rejection of claim 34. Booth further complements Shaughnessy by disclosing the step of

'performing an operation which manipulates the data in a data field associated with the at

least one date field of the database according to the converted symbolic representation of the

date' by suggesting that a string representation can be used to sort and index the converted

dates such that the dates appear in chronological order (page 945 et seq), whereby said dates,

contained in a field of the database, are sorted accordingly in a different field of the database

(page 839-40 et seq).

As to claim 45, Shaughnessy, Hazama and Booth disclose the invention as discussed in the

rejection of claim 35. Booth further complements Shaughnessy by disclosing the step of 'performing

an operation which manipulates the data ire a data field associated with the at least one date field of

the database according to the converted symbolic representation of the date' by suggesting that a

string representation can be used to sort and index the converted dates such that the dates appear in

chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are

sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 46, Shaughnessy, Ha2ama and Booth disclose the invention as discussed in the

rejection of claim 34. Shaughnessy further discloses the step of 'converting at least a substantial

portion of each of the plurality of symbolic representations of dates in the at least one date field and

repeating this step until each of the date data entries in the at least one date field is converted into the

format that does not have the ambiguity' by converting the current date stored in the database field

from an ambiguous six digit format (YYMNIDD) into an unambiguous 8-digit format (CCYYMMDD),

wherein the century for the date is specified (col. 5, lines 48-50 et seq).

As to claim 47, Shaughnessy, Hazama and Booth disclose the invention as discussed in the

rejection of claim 35. Shaughnessy further discloses the step of 'converting at least a substantial

portion of each of the plurality of symbolic representations of dates in the at least one date field and

repeating this step until each of the date data entries in the at least one date field is converted into the

format that does not have the ambiguity' by converting the current date stored in the database field

from an ambiguous six: digit format (YYMMDD) into an unambiguous 8-digit format (CCYYMMDD),

wherein the century for the date is specified (col. 5, lines 48-50 et seq).

As to claim 48, Shaughnessy, Hazama and Booth disclose the invention as discussed in the

rejection of claim 46. Booth further complements Shaughnessy by disclosing the step of 'collectively

sorting the converted symbolic representations prior to the step of running the program on the

converted symbolic representations' by suggesting that a string representation can be used to sort

and index the converted dates such that the dates appear in chronological order (page 945 et seq),

whereby said dates, contained in a field of the database, are sorted accordingly in a different field of

the database (page 839-40 et seq).

As to claim 49, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 47. Booth further complements Shaughnessy by disclosing the step of `collectively sorting the converted symbolic representations prior to the step of running the program on the converted symbolic representations' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 50, Shaughnessy, Hazama and Booth di:3close the invention as discussed in the rejection of claim 46. Booth further complements Shaughnessy and Hazama by disclosing the step of 'collectively manipulating the converted symbolic representations' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 83940 et seq).

As to claim 51, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 49. Booth further complements Shaughnessy and Hazama by disclosing the step of 'collectively manipulating the converted symbolic representations' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq) whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 52, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 46. Booth further complements '31haughnessy and Hazama by disclosing the step of 'collectively sorting the converted symbolic representations according to a different data field in the database than the at least one date field, prior to the step of running the program' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 53, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 47. Booth further complements Shaughnessy and Hazama by disclosing the step of 'collectively sorting the converted symbolic representations according to a different data field in the database than the at least one date field, prior to the step of running the program' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq), whereby said dates. contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 54, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 52. Booth further complements Shaughnessy and Hazama by disclosing the step of 'collectively manipulating the converted symbolic representations' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq.), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 83940 et seq).

As to claim 55, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 53. Booth further complements Shaughnessy and Hazama by disclosing the step of 'collectively manipulating the converted symbolic representations' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 83940 et seq).

As to claim 56, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 52. Booth further complements Shaughnessy and Hazama by disclosing the step of 'performing an operation which manipulates the data in a data field associated with the at least one date field of the database according to the converted symbolic representation of the date' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 57, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 53. Booth further complements Shaughnessy and Hazama by disclosing the step of 'performing an operation which manipulates the data in a data field associated with the at least one date field of the database according to the converted symbolic representation of the date' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq),
whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 58, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 54. Booth further complements Shaughnessy and Hazama by disclosing the step of 'performing an operation which manipulates the data in a data field associated with the at least one date field of the database according to the converted symbolic representation of the date' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq., whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 59, Shaughnessy, Hazama and Booth disclose the invention as discussed in the rejection of claim 55. Booth further complements Shaughnessy and Hazama by disclosing the step of 'performing an operation which manipulates the data in a data field associated with the at least one date field of the database according to the converted symbolic representation of the date' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

19.   Claim 60 is rejected under 35 U S.C. 103(a) as being unpatentable over Shaughnessy in view of Hazama, and further in view of Booth.

As to claim 60, Shaughnessy substantially discloses the invention including the claimed 'method for representing and utilizing dates stored in at least one date field of a database utilizing symbolic representations of the dates stored in the at least one date field of the database, which are in a format that creates ambiguity between dates in each of a pair of adjacent centuries' as a method

and system for modifying and operating a computer system to perform operations on date fields

having a two digit representation for the year without mistaking the year 2000 and the year 1900 (col.

1, lines 7-14 et seq). Shaughnessy also suggests the conversion of all dates within the database from

a two digit format to a four digit format as a viable, but costly alternative for the year 2000 problem

(col. 1, lines 31-46 et seq). The method as recited in the body of the claim is -particularly taught by

the cited references as follows:

Shaughnessy discloses the steps of converting each of the symbolic representations of dates stored

in the at least one date field of the database to a symbolic representation of each of the respective

dates that does not create the ambiguity, by windowing the symbolic representations of each of the

respective dates as stored in the at least one date field of the database against a pivot year

represented by one of the symbolic representations of the dates as stored in the at least one date

field of the database, without modifying any of the symbolic representations of dates in the at least

one date field of the database for purposes of such windowing and converting; and running a program

on each of the converted symbolic representations of each of the respective dates to manipulate data

in the database according to the dates represented by the converted symbolic representations,

separately from the date data symbolic representations of dates contained in the at least one date

field of the database' by providing a database having a 6 digit-field for storing a Date type in the form

of MMDDYY(see appendix in col. 18, Date Type A), wherein the MM represents the month, DD

represents the day and YY represents the year for a particular six digit-date falling within a 10-decade

period of time' as a date having a cycle or a range of a 100 years (col. 18, Cycle/Range C1 = THE

DATE CYCLE IS 100 YEARS). Shaughnessy further provides a subroutine for determining the

current date 16 to thereby select a 100 year cycle wherein the current date is the pivot date and

wherein the cycle end a 100 years from said current date (col.5, lines 31-36). Additionally,

Shaughnessy discloses the comparison of the current date to the date when the system was installed

with the modifications (modified system install date) to thereby determine the century value (col.5,

lines 36-65 et seq). Alternatively, Shaughnessy provides a subroutine that compares the YYMMDD

portions' of the date to the corresponding date portion at the end of the 100 year cycle to thereby

determine the century value (col.7, lines 7-15 et seq). Finally, Shaughnessy provides a subroutine for

appending the determined century value before the YYMMDD date in order to yield a CCYYMMDD

'6The current date, by virtue of being the pivot date ii the 100 year-cycle and by being initially set to

the operating system date, initially set to 0000 (see col. 7, lvies 1'9-20 et seq), is therefore not any

later than the earliest date in 100 year-cycle in the database.

" Shaughnessy specifically suggests that it might be desirable to set the current date to a date which

compares low to all other dates (col. 7, lines 16-17 et seq). date format, whereby said appending was

performed without modifying any of the date fields stored in the database (col.5, lines 46-51; col.6,

lines 57-65 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest Y 1 Y2 year designator in the database in the selection of the 10 decade

window. However, Hazama discloses an analogous date processing system having two digit dates

spanning from the 20th century to the 21st century stored in memory (page 2, claim 1) wherein, for a

100 year window (restricting the all the dates in the database between the 20th and 21st centuries),

the pivot date for the window is selected based on a two digit year date that is smaller than the

smallest two digit year date in the database (e.g. if smallest two digit date stored in database is 73,

the pivot date for the 100 year window is chosen to be 72). (page 4 of translated document, last

paragraph). It would have been obvious to one of ordinary skill in the art of data processing to

combine the teachings of the cited references. The ordinary skilled artisan having read Shaughnessy

would immediately see the need to determine which 100 year span to use. This determination would

have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being

smaller than the smallest two digit date in the database having all the dates within a 100 year period

as a solution to restrict the selection of Shaughnessy's window and thereby forcing all dates already

stored in the database to fall in the 20th century.

Shaughnessy and Hazama do not specifically disclose the sorting of converted dates in the form

$C.sub. 1 C.sub.2 Y.sub. 1 Y.sub.2 M.sub. 1 M.siub.2 D.sub. 1 D.sub.2$. However, Booth discloses an

analogous system that utilizes the Clipper programming language to process dates stored in a

database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq.

In particular, analogously to Shaughnessy, Booth discloses the SET EPOCH command for comparing

an entered two digit date with the year digit (pivot date) of the epoch setting to determine the century

to place the date into (p. 941, see SET EPOCH paragraph et seq) thereby converting a six digit date

(MMDDYY) into a corresponding eight digit date (C 1 C2Y 1 Y2M 1 M2D 1 D2) (see p. 940--941).

Additionally, Booth complements Shaughnessy by suggesting the sorting of converted dates after

having been reformatted by the SET EPOCH command (p. 945). It would have been obvious to one

of ordinary skill in the art of data processing at the time of the instant invention to combine the

teachings of the cited references. Booth's teaching of sorted reformatted dates would facilitate the

Shaughnessy Hazama's system to return the reformatted dates in chronological sequence. Such

disclosed sorting would therefore be very useful when indexing the database in date order, as

suggested by Booth on page 945.

20. Claim 61 is rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view of

Hazama, and further in view of Booth.

As to claim 61, Shaughnessy substantially discloses the invention including the claimed

`method for representing and utilizing dates stored in at least one date field of a database utilizing

symbolic representations of the dates stored in the at least one date field of the database, which are

in a format that creates ambiguity between dates in each of a pair of adjacent centuries' as a method

and system for modifying and operating a computer system to perform operations on date fields

having a two digit representation for the year without mistaking the year 2000 and the year 1900 (col.

1, lines 7-14 et seq). Shaughnessy also suggests the conversion of all dates within the database from

a two digit format to a four digit format as a viable, but costly alternative for the year 2000 problem

(col. 1, lines 31-46 et seq). The method as recited in the body of the claim is particularly taught by the

cited references as follows:

Shaughnessy discloses the steps of converting each of the symbolic representations of dates stored

in the at least one date field of the database to a symbolic representation of each of the respective

dates that does not create the ambiguity, by windowing the symbolic representations of each of the

respective dates as stored in the at least one date field of the database against a pivot year

represented by one of the symbolic representations of the dates as stored in the at least one date

field of the database, without modifying any of the symbolic representations of dates in the at least

one date field of the database for purposes of such windowing and converting; and running a program

collectively on each of the converted symbolic representations of each of the respective dates to

manipulate the dates represented by the converted symbolic representations, separately from the

symbolic representations of dates contained in the at least one date field of the database' by

providing a database having a 6 digit-field for storing a Date type in the form of MMDDYY (see

appendix in col. 18, Date Type A), wherein the MM represents the month, DD represents the day and

YY represents the year for a particular six digit-date falling within a 10-decade period of time' as a

date having a cycle or a range of a 100 years (col. 18, Cycle/Range CI = THE DATE CYCLE IS 100

YEARS). Shaughnessy further provides a subroutine for determining the current date 18 to thereby

select a 100 year cycle wherein the current date is the pivot date and wherein the cycle ends a 100

years from said current date (co1.5, lines 31-36). Additionally, Shaughnessy discloses the

comparison of the current date to the date when the system was installed with the modifications

(modified system install date) to thereby determine the century value (col.5, lines 36-65 et seq).

Alternatively, Shaughnessy provides a subroutine that compares the YYMMDD portion 19 of the date

to the corresponding date portion at the end of the 100 year cycle to thereby determine the century

value (col.7, lines 7-15 et seq). Finally, Shaughnessy provides a subroutine for appending the

determined century value before the YYMMDD date in order to yield a CCYYMMDD date format,

whereby said appending was performed without modifying any of the date fields stored in the

database (col.5, lines 46-51; co1.6, lines 57-65 et seq). Shaughnessy does not particularly detail that

the YaYb value for the first decade of the window is no later than the earliest Y I Y2 year designator in

the database in the selection of the 10 decade window. However, Hazama discloses an analogous

date processing system having two digit dates spanning from the 20th century to the 21st century

stored in memory (page 2, claim 1) wherein, for a 100 year window (restricting the all the dates in the

database between the 20th and 21st centuries), the pivot date for the window is selected based on a

two digit year date that is smaller than the smallest two digit year date in the database (e.g. if

"The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set to

the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any

later than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date, which

compares low to all other dates (col. 7, lines 16-17 et seq).

smallest two digit date stored in database is 73, the pivot date for the 100 year window is

chosen to be 72). (page 4 of translated document, last paragraph). It would have been

obvious to one of ordinary skill in the art of data processing to combine the teachings of the

cited references. The ordinary skilled artisan having read Shaughnessy would immediately

see the need to determine which 100 year span to use. This determination would have led the

ordinary skilled artisan to the Hazama reference, which teaches the pivot date being smaller than the

smallest two digit date in the database having all the dates within a 100 year period as a solution to

restrict the selection of Shaughnessy's window and thereby forcing all dates already stored in the

database to fall in the 20th century.

Shaughnessy and Hazama do not specifically disclose the sorting of converted dates in the form C.

sub. l C.sub.2 Y. sub. l Y.sub.2 M. sub. l M.sub.2. D. sub. l D.sub.2. However, Booth discloses an

analogous system that utilizes the Clipper programming language to process dates stored in a

database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq. In particular, analogously

to Shaughnessy, Booth discloses the SET EPOCH command for comparing an entered two digit date

with the year digit (pivot date) of the epoch setting to determine the century to place the date into (p.

941, see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a

corresponding eight digit date (ClC2YlY2MlM2DlD2) (see p. 940-941). Additionally, Booth

complements Shaughnessy by suggesting the sorting of converted dates after having been

reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary

skill in the art of data processing at the time of the instant invention to combine the teachings of the

cited references. Booth's teaching of sorted reformatted dates would facilitate Shaughnessy's

system to return the reformatted dates in chronological sequence. Such disclosed sorting

would therefore be very useful when indexing the database in date order, as suggested by

Booth on page 945.

21. Claim 62 is rejected under 35 U. S. C. 103 (a) as being unpatentable over Shaughnessy in

view of Hazama, and further in view of Booth.

As to claim 62, Shaughnessy substantially discloses the invention including the claimed

method for representing and utilizing dates stored in at least one date field of a database

utilizing symbolic representations of the dates stored in the at least one date field of the

database, which are in a format that creates ambiguity between dates in each of a pair of

adjacent centuries as a method and system for modifying and operating a computer system to

perform operations on date fields having a two digit representation for the year without

mistaking the year 2000 and the year 1900 (col.1, lines 7-14 et seq). Shaughnessy also

suggests the conversion of all dates within the database from a two digit format to a four digit format

as a viable, but costly alternative for the year 2000 problem (col. 1, lines 31-46 et seq). The method

as recited in the body of the claim is particularly taught by the cited references as follows:

Shaughnessy discloses the steps of `converting each of the symbolic representations of dates stored

in the at least one date field of the database to a symbolic representation of each of the respective

dates that does not create the ambiguity, by windowing the symbolic representations of each of the

respective dates as stored in the at least one date field of the database against a pivot year

represented by one of the symbolic representations of the dates as stored in the at least one date

field of the database, without the addition of any new data field to the database for purposes of such

windowing and converting; and running a program on the stored converted symbolic representations

to manipulate data in the database according to the dates represented by the converted symbolic

representations, separately from the symbolic representations of-dates--contained in the at least one

date field of the database' by providing a database having a 6 digit-field for storing a Date type in the

form of M1VMDYY(see appendix in col. 18, Date Type A), wherein the MM represents the month, DD

represents the day and YY represents the year for a particular six digit-date falling within a 10-decade

period of time' as a date having a cycle or a range of a 100 years (col. 18, Cycle/Range CI = THE

DATE CYCLE IS 100 YEARS). Shaughnessy further provides a subroutine for determining the

current date to thereby select a 100 year cycle wherein the current date is the pivot date and wherein

the cycle ends a 100 years from said current date (co1.5, lines 31-36). Additionally, Shaughnessy

discloses the comparison of the current date to the date when the system was installed with the

modifications (modified system install date) to thereby determine the century value (col.5, lines 36-65

et seq). Alternatively, Shaughnessy provides a subroutine that compares the (YYMMDD portion 21 of

the date to the corresponding date portion at the end of the 100 year cycle to thereby determine the

century value (col.7, lines 7-15 et seq). Finally, Shaughnessy provides a subroutine for appending

20The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set to

the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any

later than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date, which

compares low to all other dates (col. 7, lines 16-17 et seq). the determined century value before the

YYMIVIDD date in order to yield a CCYYMMDD date format, whereby said appending was performed

without modifying any of the date fields stored in the database (col.5, lines 46-51; col.6, lines 57-65 et

set.

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest Y I Y2 year designator in the database in the selection of the 10 decade

window. However, Hazama discloses an analogous date processing system having two digit dates

spanning from the 20th century to the 21st century stored in memory (page 2, claim 1) wherein, for a

100 year window (restricting the all the dates in the database between the 20th and 21st centuries),

the pivot date for the window is selected based on a two digit year date that is smaller than the

smallest two digit year date in the database (e.g. if smallest two digit date stored in database is 73,

the pivot date for the 100 year window is chosen to be 72). (page 4 of translated document, last

paragraph). It would have been obvious to one of ordinary skill in the art of data processing to

combine the teachings of the cited references. The ordinary skilled artisan having read Shaughnessy

would immediately see the need to determine which 100 year span to use. This determination would

have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being

smaller than the smallest two digit date in the database having all the dates within a 100 year period

as a solution to restrict the selection of Shaughnessy's window and thereby forcing all dates already

stored in the database to fall in the 20th century.

Shaughnessy and Hazama do not specifically disclose the sorting of converted dates in the form

C.sub. I C.sub.2 Y.sub. I Y.sub.2 M.sub.I M.sub.2 D.sub. I D.sub.2. However,

Booth discloses an analogous system that utilizes the Clipper programming language to process

dates stored in a database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq. In

particular, analogously to Shaughnessy, Booth discloses the SET EPOCH command for comparing

an entered two digit date with the year digit (pivot date) of the epoch setting to determine the century

to place the date into (p. 941, see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a corresponding eight digit date (CIC2YIY2MIM2DID2) (see p. 940-941). Additionally, Booth complements Shaughnessy by suggesting the sorting of converted dates after having been reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary skill in the art of data processing at the time of the instant invention to combine the teachings of the cited references. Booth's teaching of sorted reformatted dates would facilitate the Shaughnessy-Hazama's system to return the reformatted dates in chronological sequence. Such disclosed sorting would therefore be very useful when indexing the database in date order, as suggested by Booth on page 945.

Shaughnessy and Hazama do not specifically disclose the step of storing the converted symbolic representations separate from the at least one date field of the database.' Booth, however, further complements the cited references by renaming and storing sorted dates in the CUSTMER.DBF NEW CUST.D73F databases (p. 841). It would have been obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited references. Booth's teaching of the storing the sorted reformatted dates would users of the Shaughnessy-Hazama's system to readily retrieve the reformatted dates in their chronological sequence at any time and in accordance with their needs.

22. Claim 63 is rejected under 35 L.S.C. 103(a) as being unpatentable over Shaughnessy in view of Hazama, and further in view of Booth.

As to claim 63, Shaughnessy substantially discloses the invention including the claimed method for representing and utilizing dates stored in at least one date field of a database utilizing symbolic representations of the dates stored in the at least one date field of the database, which are

in a format that creates ambiguity between dates in each of a pair of adjacent centuries' as a method

and system for modifying and operating a computer system to perform operations on date fields

having a two digit representation for the year without mistaking the year 2000 and the year 1900

(col.1, lines 7-14 et seq). Shaughnessy also suggests the conversion of all dates within the database

from a two digit format to a four digit format as a viable, but costly alternative for the year 2000

problem (col. 1, lines 31-46 et seq). The method as recited in the body of the claim is particularly

taught by the cited references as follows:

Shaughnessy discloses the steps of `converting each of the symbolic representations of' dates stored

in the at least one date field of the database to a symbolic representation of each of the respective

dates that does not create the ambiguity, by windowing the symbolic representations of each of the

respective dates as stored in the at least one date field of the database against a pivot year

represented by one of the symbolic representations of the dates as stored in the at least one date

field of the database, without the addition of any new data field to the database for purposes of such

windowing and converting; and running a program collectively on the stored converted symbolic

representations to manipulate the dates represented by the converted symbolic representations,

separately from the symbolic representations of dates contained in the at least one date field of the

database' by providing a database having a 6 digit-field for storing a Date type, in the forth of

MMDDYY(see appendix in col. 18, Date Type A), wherein the MM represents the month, DD

represents the day and YY represents the year for a particular six digit.-date falling within a

10-decade period of time' as a date having a cycle or a range of a 1.00 years (col. 18, Cycle/Range C

1 =THE DATE CYCLE IS 100 YEARS) Shaughnessy further provides a subroutine for

determining the current date 22 to thereby select a 100 year cycle wherein the current date is

the pivot date and wherein the cycle ends a 100 years from said current date (co1.5, lines 31

36). Additionally, Shaughnessy discloses the comparison of the current date to the date

when the system was installed with the modifications (modified system install date) to

thereby determine the century value (co1.5, lines 36-65 et seq). Alternatively, Shaughnessy

provides a subroutine that compares the YYMMDD portion 23 of the date to the corresponding date

portion at the end of the 100 year cycle to thereby determine the century value (co1.7, lines 7-15 et

seq). Finally, Shaughnessy provides a subroutine for appending the determined century value before

the YYMMDD date in order to yield a CCYYMMDD. The current date, by virtue of being the pivot

date H the 100 year-cycle and by being initially set to the operating system date, initially set to 0000

(see col. 7, lines 19-20 et seq), is therefore not any later than the earliest date in 100 year-cycle in the

database. ·

Shaughnessy specifically suggests that it might be desirable to set the current date to a date, which

compares low to all other dates (col. 7, lines 16-17 et seq). date format, whereby said appending was

performed without modifying any of the date fields stored in the database (col.5, lines 46-51; col.6,

lines 17-65 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest Y 1 Y2 year designator in the database in the selection of the 10 decade

window. However, Hazama discloses an analogous date processing system having two digit dates

spanning from the 20th century to the 21st century stored in memory (page 2, claim 1) wherein, for a

100 year window (restricting the all the dates in the database between the 20th and 21st centuries),

the pivot date for the window is selected based on a two digit year date that is smaller than the

smallest two digit year date in the database (e.g. if smallest two digit date stored in database is 73,

the pivot date for the 100 year window is chosen to be 72). (page 4 of translated document, last

paragraph). It would have been obvious to one of ordinary skill in the art of data processing to

combine the teachings of the cited references. The ordinary skilled artisan having read Shaughnessy

would immediately see the need to determine which 100 year span to use. This determination would

have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being

smaller than the smallest two digit date in the database having all the dates within a 100 year period

as a solution to restrict the selection of Shaughnessy's window and thereby forcing all dates already

stored in the database to fall in the 20th century.

Shaughnessy and Hazama do not specifically disclose the sorting of converted dates in the form

$C_1 C_2 Y_1 Y_2 M_1 M_2 D_1 D_2$. However, Booth discloses an

analogous system that utilizes the Clipper programming language to process dates stored in a

database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq.)

In particular, analogously to Shaughnessy, Booth discloses the SET EPOCH command for comparing

an entered two digit date with the year digit (pivot date) of the epoch setting to determine the century

to place the date into (p. 941, see SET EPOCH paragraph et seq) thereby converting a six digit date

(M'VIDDYY) into a corresponding eight digit date (C 1 C2Y 1 Y2M 1 M2D 1 D2) (see p. 940-941).

Additionally, Booth complements Shaughnessy by suggesting the sorting of converted dates after

having been reformatted by the SET EPOCH command (p. 945). It would have been obvious to one

of ordinary skill in the art of data processing at the time of the instant invention to combine the

teachings of the cited references. Booth's teaching of sorted reformatted dates would facilitate the

Shaughnessy Hazama's system to return the reformatted dates in chronological sequence. Such

disclosed sorting would therefore be very useful when indexing the database in date order, as

suggested by Booth on page 945.

Shaughnessy and Hazama do not specifically disclose the step of 'storing the converted symbolic

representations separate from the at least one date field of the database.' Booth, however, further

complements the cited references by renaming and storing sorted dates in the CUSTMER.DBF NEW
CUST.DBF databases (p. 841). It would have been obvious to one of ordinary skill in the art of data
processing to combine the teachings of the cited references. Booth's teaching of the storing the
sorted reformatted dates would users of the Shaughnessy- Hazama's system to readily retrieve the
reformatted dates in their chronological sequence at any time and in accordance with their needs.

23. Claim 64 is rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view of
Hazama, and further in view of Booth.

As to claim 64, Shaughnessy substantially discloses the invention including the claimed
'method for representing and utilizing dates stored in at least one date field of a database utilizing
symbolic representations of the dates stored in the at least one date field of the database, which are
in a format that creates ambiguity between dates in each of a pair of adjacent centuries' as a method
and system for modifying and operating a computer system to perform operations on date fields
having a two digit representation for the year without mistaking the year 2000 and the year 1900
(col.1, lines 7-14 et seq). Shaughnessy also suggests the conversion of all dates within the database,
from a two digit format to a four digit format as a viable, but costly alternative for the year 2000
problem (col. 1, lines 31-46 et seq). The method as recited in the body of the claim is particularly
taught by the cited references as follows:
Shaughnessy discloses the steps of 'converting each of the symbolic representations of dates stored
in the at least one date field of the database to a symbolic representation of each of the respective
dates that does not create the ambiguity, by windowing the symbolic representations of each of the
respective dates as stored in the at least one date field of the database against a pivot year

represented by one of the symbolic representations of the dates as stored in the at least one date

field of the database, without modifying any of the symbolic representations of dates in the at least

one date field of the database for purposes of such windowing and converting; and running a program

on the stored converted symbolic representations to manipulate data in the database according to the

dates represented by converted symbolic representations, separately from the; symbolic

representations of dates contained in the at least one date field of the database' by providing a

database having a 6 digit-field for storing a Date type in the form of MMDDYY(see appendix in col.

18, DateType A), wherein the MM represents the month, DD represents the day and YY represents

the year for a particular six digit-date falling within a 10-decade period of time' as a date

having a cycle or a range of a 100 years (col. 18, Cycle/Range C I = THE DATE CYCLE I S

100 YEARS). Shaughnessy further provides a subroutine for determining the current date`"

to thereby select a 100 year cycle wherein the current date is the pivot date and wherein the

cycle ends a 100 years from said current date (co1.5, lines 31-36). Additionally, Shaughnessy

discloses the comparison of the current date to the date when the system was installed with the

modifications (modified system install date) to thereby determine the century value (col.5, lines 36-65

et seq). Alternatively, Shaughnessy ;provides a subroutine that compares the YYMMDD portion 25 of

the date to the corresponding date portion at the end of the 100 year cycle to thereby determine the

century value (co1.7, lines 7-15 et seq). Finally, Shaughnessy provides a subroutine for appending

the determined century value before the YYMMDD date in order to yield a CCYYMMDD date format,

whereby said appending was performed without modifying any of the date fields stored in the

database (col.5, lines 46-51; col.6, lines 57-65 et seq).

24 The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set to

the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any

later than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date, which

compares low to all other dates (col. 7, lines 16-17 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest YIY2 year designator in the database in the selection of the 10 decade window.

However, Hazama discloses an analogous date processing system having two digit dates spanning

from the 20th century to the 21st century stored in memory (page 2, claim 1) wherein, for a 100 year

window (restricting the all the dates in the database between the 20th and 21st centuries), the pivot

date for the window is selected based on a two digit year date that is smaller than. the smallest two

digit year date in the database (e.g. if smallest two digit date stored in database is 73, the pivot date

for the 100 year window is chosen to be 72). (page 4 of translated document, last paragraph). It

would have been obvious to one of ordinary skill in the art of data processing to combine the

teachings of the cited references. The ordinary skilled artisan having read Shaughnessy would

immediately see the need to determine which 100 year span to use. This determination would have

led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being smaller

than the smallest two digit date in the database having all the dates within a 100 year period as a

solution to restrict the selection of Shaughnessy's window and thereby forcing all dates already stored

in the database to fall in the 20th century.

Shaughnessy and Hazama do not specifically disclose the sorting of converted dates in the form

C.sub. I C.sub.2 Y.sub. I Y.sub.2 M.sub. I M.sub.2 D.sub. I D.sub.2. However, Booth discloses an

analogous system that utilizes the Clipper programming language to process dates stored in a

database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq). In particular, analogously

to Shaughnessy, Booth discloses the SET EPOCH command for comparing an entered two digit date

with the year digit (pivot date) of the epoch setting to determine the century to place the date into (p.

941, see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a

corresponding eight digit date (CIC2YIY2MIM2DID2) (seep. 940-941). Additionally, Booth

complements Shaughnessy by suggesting the sorting of converted dates after having been

reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary

skill in the art of data processing at the time of the instant invention to combine the teachings of the

cited references. Booth's teaching of sorted reformatted dates would facilitate the Shaughnessy-

Hazama's system to return the reformatted dates in chronological sequence. Such disclosed sorting

would therefore be very useful when indexing the database in date order, as suggested by Booth on

page 945.

Shaughnessy and Hazama do not specifically disclose the step of 'storing the converted symbolic

representations separate from the at least one date field in the database.' Booth, however, further

complements the cited references by renaming and storing sorted dates in the CUSTMER.DBF NEW

CUST.DBF databases (p. 841). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. Booth's teaching of the storing the

sorted reformatted dates would users of the Shaughnessy-Hazama's system to readily retrieve the

reformatted dates in their chronological sequence at any time and in accordance with their needs.

24. Claim 65 is rejected under 35 U.S C. 103(a) as being unpatentable over Shaughnessy in view of

Hazama, and further in view of Booth.

As to claim 65, Shaughnessy substantially discloses the invention including the claimed

method for representing and utilizing dates stored in at least one date field of a database

utilizing symbolic representations of the dates stored in the at least one date field of the

database, which are in a format that creates ambiguity between dates in each of a pair of

adjacent centuries' as a method and system for modifying and operating a computer system

to perform operations on date fields having a two digit representation for the year without

mistaking the year 2000 and the year 1900 (col.1, lines 7-14 et seq). Shaughnessy also

suggests the conversion of all dates within the database from a two digit format to a four digit format

as a viable, but costly alternative for the year 2000 problem (col. 1, lines 31-46 et seq). The method

as recited in the body of the claim is particularly taught by the cited references as follows:

Shaughnessy discloses the steps of converting each of the symbolic representations of dates stored

in the at least one date field of the database to a symbolic representation of each of the respective

dates that does not create the ambiguity, by windowing the symbolic representations of each of the

respective dates as stored in the at least one date field of the database against a pivot year

represented by one of the symbolic representations of the dates as stored in the at least one date

field of the database, without modifying any of the symbolic representations of dates in the at least.

one date field of the database for purposes of such windowing and converting; and running a program

collectively on the stored converted symbolic representations to manipulate the dates represented by

the converted symbolic representations, separately from the symbolic representations of dates

contained in the at least one date field of the database' by providing a database having a 6 digit-field

for storing a Date type in the form of MMDDYY(see appendix in col. 18, Date Type A), wherein the

MM represents the month, DD represents the day and YY represents the year for a particular six

digit-date falling within a 10-decade period of time' as a date having a cycle or a range of a 100 years

(col. 18, Cycle/Range C 1 = THE DATE CYCLE IS 100 YEARS). Shaughnessy further provides a

subroutine for determining the current date26 to thereby select a 100 year cycle wherein the current

date is the pivot date and wherein the cycle ends a 100 years from said current. date (col.5, lines

31-36). Additionally, Shaughnessy discloses the comparison of the current date to the date when the

system was installed with the modifications (modified system install date) to thereby determine the

century value (col.5, lines 36-65 et seq). Alternatively, Shaughnessy provides a subroutine that

compares the YYMMDD portion27 of the date to the corresponding date portion at the end of the 100

year cycle to thereby determine the century value (col.7, lines 7-15 el. seq). Finally, Shaughnessy

provides a subroutine for appending the determined century value before the YYMMDD date in order

to yield a CCYYMMDD date format, whereby said appending was performed without modifying any of

the date fields stored in the database (co1.5, lines 46-51; col.6, lines 57-65 et seq). Shaughnessy

does not particularly detail that the YaYb value for the first decade of the window is no later than the

earliest YI Y2 year designator in the database in the selection of

The current date, by virtue of being the pivot date in the 1.00 year-cycle and by being initially set to

the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any

later than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date, which

compares low to all other dates (col. 7, lines 16-17 et seq.).

the 10 decade window. However, Hazama discloses an analogous date processing system having

two digit dates spanning from the 20th century 1:o the 21st century stored in memory (page 2, claim

1) wherein, for a 100 year window (restricting the all the dates in the database between the 20th and

21st centuries), the pivot date for the window is selected based on a two digit year date that is smaller

than the smallest two digit year date in the database (e.g. if smallest two digit date stored in database

is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated document, last

paragraph). It would have been obvious to one of ordinary skill in the art of data processing to

combine the teachings of the cited references. The ordinary skilled artisan having read Shaughnessy

would immediately see the need to determine which 100 year span to use. This determination would

have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being

smaller than the smallest two digit date in the database having all the dates within a 100 year period

as a solution to restrict the selection of Shaughnessy's window and thereby forcing all dates already

stored in the database to fall in the 20th century.

Shaughnessy and Hazama do not specifically disclose the sorting of converted dates in the form C_1

C_2 Y_1 Y_2 M_1 M_2 D_1 D_2. However, Booth discloses an

analogous system that utilizes the Clipper programming language to process dates stored in a

database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq. In particular, analogously

to Shaughnessy, Booth discloses the SET EPOCH command for comparing an entered two digit date

with the year digit (pivot date) of the epoch setting to determine the century to place the date into (p.

941, see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a

corresponding eight digit date (C1 C2Y l Y2M 1 M2D 1 D2) (see p. 940-941). Additionally, Booth

complements Shaughnessy by suggesting the sorting of converted dates after having been

reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary

skill in the art of data processing at the time of the instant invention to combine the teachings of the

cited references. Booth's teaching of sorted reformatted dates would facilitate the Shaughnessy-

Hazama's system to return the reformatted dates in chronological sequence. Such disclosed sorting

would therefore be very useful when indexing the database in date order, as suggested by Booth on

page 945.

Shaughnessy and Hazama do not specifically disclose the step of 'storing the converted symbolic

representations separate from the at least one date field in the database.' Booth, however, further

complements the cited references by renaming and storing sorted dates in the CUSTMER.DBF NEW

CUST.DBF databases (p. 841). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. Booth's teaching of the storing the

sorted reformatted dates would users of the Shaughnessy-Hazama's system to readily retrieve the

reformatted dates in their chronological sequence at any time and in accordance with their needs.

25. Claim 66 is rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view of

Hazama.

As to claim 66, Shaughnessy substantially discloses the invention including the claimed

Shaughnessy discloses the claimed 'method of processing dates in a database' as a method

and system for modifying and operating a computer system to perform operations on date

fields having a two digit representation for the year without mistaking the year 2000 and the

year 1900 (col.1, lines 7-14 et seq). Shaughnessy also suggests the conversion of all dates

within the database from a two digit format to a four digit format as a viable, but costly

alternative for the year 2000 problem (col. 1, lines 31-46 et seq). The method as recited in

the body of the claim is particularly taught by the cited references as follows:

Shaughnessy discloses the step of 'providing a database with dates stored in at least one

date field therein according to a format wherein M 1 M2 is the numerical month designator,

DI D2 is the numerical day designator, and YI Y2 is the numerical year designator' as a

database having a 6 digit-field for storing a Date type in the form of MMDDYY(see

appendix in col. 18, Date Type A), wherein the MM represents the month, DD represents the

day and YY represents the year for a particular six digit-date falling within a 10-decade

period of time' as a date having a cycle or a range of a 100 years (col. 18, Cycle/Range C 1

THE DATE CYCLE IS 100 YEARS). As pointed out in column 2, lines 11-14 and column

3, lines 4-8 of Patent No. 5,806,063, all dates in commercial and industrial databases span

within one 100 year. Shaughnessy's system being of the commercial or industrial kind

described in the cited patent, must therefore, as a practical matter, inherently incorporate this

limitation. Shaughnessy also discloses a subroutine for determining the current date 28 to

thereby select a 100 year cycle wherein the current date is the pivot date and wherein the

cycle ends a 100 years from said current date (col.5, lines 31-36). Additionally, Shaughnessy

The current date, by virtue of being the pivot date iii the 100 year-cycle and by being initially set to the

operating system date, initially set to 0000 (see col. 7, lies 19-20 et seq), is therefore not any later

discloses the step of 'determining a century designator CI C2 for each date in the database,

CI C2 having a first value if Y1 Y2 is less than YAYB and having a second value if Y1 Y2

is equal to or greater than YAYB' as the comparison of the current date to the date when the

system was installed with the modifications (modified system install date) to thereby

determine the century value (col.5, lines 36-65 et seq). Alternatively, Shaughnessy discloses

the comparison of the YYMMDD portion 29 of the date to the corresponding date portion at

the end of the 100 year cycle to thereby determine the century value (col.7, lines 7-15 et seq).

Finally, Shaughnessy discloses the step of 'reformatting the symbolic representation of each

symbolic representation of a date in a portion of the at least one date field in the database,

without the addition of any new data field to the database, with the reformatted symbolic

representation of each date in the database having the values C I C2, Y 1 Y2, M 1 M2, and

D I D2; and repeating the step of reformatting until each symbolic representation of a date in

the at least one date field has been reformatted in order to facilitate collectively further

processing the reformatted symbolic representations of each of the symbolic representations

of each of the dates' by appending the determined century value before the YYMMDD date

in order to yield a CCYYMMDD date format, whereby said appending was performed

without adding a new field to the database (col.5, lines 46-51; co1.6, lines 57-65 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the

window is no later than the earliest YlY2 year designator in the database in the selection of

than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date

which compares low to all other dates (col. 7, lines 16-17 et seq).

the 10 decade window. However, Hazama discloses an analogous date processing system having

two digit dates spanning from the 20th century to the 21st century stored in memory (page 2, claim 1)

wherein, for a 100 year window (restricting the all the dates in the database between the 20th and

21st centuries), the pivot date for the window is selected based on a two digit year date that is smaller

than the smallest two digit year date in the database (e.g. if smallest two digit date stored in database

is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated document, last

paragraph). It would have been obvious to one of ordinary skill in the art of data processing to

combine the teachings of the cited references. The ordinary skilled artisan having read Shaughnessy

would immediately see the need to determine which 100 year span to use. This determination would

have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being

smaller than the smallest two digit date in the database having all the dates within a 100 year period

as a solution to restrict the selection of Shaughnessy's window and thereby forcing all dates already

stored in the database to fall in the 20th century.

26.   Claim 67 is rejected under 35 U.S.C. 103(a) as obvious over Shaughnessy in view of Hazama.

As to claim 67, Shaughnessy substantially discloses the invention including the claimed `method of

processing dates in a database' as a method and system for modifying and operating a computer

system to perform operations on date fields having a two digit representation for the year without

mistaking the year 2000 and the year 1900 (col.1, lines 7-14 et seq). Shaughnessy also suggests the

conversion of all dates within the database from a two digit format to a four digit format as a viable,

but costly alternative for the year 2000 problem (col. 1, lines 31-46 et seq). The method as recited in

the body of the claim is particularly taught by the cited references as follows:

Shaughnessy discloses the step of `providing a database with dates stored in at least one

date field therein according to a format wherein Y 1 Y2 is the numerical year designator' as a

database having a 6 digit-field for storing a Date type in the form of MMDDYY(see appendix in col.

18, Date Type A), wherein the MM represents the month, DD represents the day and YY represents

the year for a particular six digit-date falling within a 10-decade period of time' as a date having a

cycle or a range of a 100 years (col. 18, Cycle/Range C 1 = THE DATE CYCLE 1S 100 YEARS). As

pointed out in column 2, lines 11-14 and column 3, lines 4-8 of Patent No. 5,806,063, all dates in

commercial and industrial databases span within one 100 year. Shaughnessy's system being of the

commercial or industrial kind described in the cited patent, must therefore, as a practical matter,

inherently incorporate this limitation. Shaughnessy also discloses a subroutine for determining the

current date 30 to thereby select a 100 year cycle wherein the current date is the pivot date and

wherein the cycle ends in 100 years from said current date (col.5, lines 31-36). Additionally,

Shaughnessy discloses the step of `determining a century designator C1 C2 for each date in the

database, CI C2 having a first value if YIY2 is less than YAYB and having a second

"The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set to

the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any late

value if YI Y2 is equal to or greater than YAYB' as the comparison of the current date to the date

when the system was installed with the modifications (modified system install date) to thereby

determine the century value (col.5, lines 36-65 et seq). Alternatively, Shaughnessy discloses the

comparison of the YYMMDD portion 31 of the date to the corresponding date portion at the end of the

100 year cycle to thereby determine the century value (col.7, lines 715 et seq). Finally, Shaughnessy

discloses the step of 'reformatting the symbolic representation of each symbolic representation of a

date in a portion of the at least one date field in the database, without the addition of any new data

field to the database, with the reformatted symbolic representation of each date in the database

having the values CI C2, YI Y2; and repeating the step of reformatting until each symbolic

representation of a date in the at least one date field has been reformatted in order to facilitate

collectively further processing the reformatted symbolic representations of each of the symbolic

representations of each of the dates' by appending the determined century value before the

YYMMDD date in order to yield CCYYMMDD date format, whereby said appending was performed

without adding a new field to the database (col.5, lines 46-51; col.6, lines 57-65 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest YIY2 year designator in the database in the selection of' the 10 decade window.

However, Hazama discloses are analogous date processing system having two digit dates spanning

from the 20th century to the 21st century stored in memory than the earliest date in 100 year-cycle in

the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date

which compares low to all other dates (col. 7, lines 16-17 et seq).

(page 2, claim 1) wherein, for a 100 year window (restricting the all the dates in the database

between the 20th and 21st centuries), the pivot date for the window is selected based on a two digit

year date that is smaller than the smallest two digit year date in the database (e.g. if smallest two digit

date stored in database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of

translated document, last paragraph). It would have been obvious to one of ordinary skill in the art of

data processing to combine the teachings of the cited references. The ordinary skilled artisan having

read Shaughnessy would immediately see the need to determine which 100 year span to use. This

determination would have led the ordinary skilled artisan to the Hazama reference, which teaches the

pivot date being smaller than the smallest two digit date in the database having all the dates within a

100 year period as a solution to restrict the selection of Shaughnessy's window and thereby forcing all

dates already stored in the database to fall in the 20th century.

27.   Claim 68 is rejected under 35 U.S.C. as being unpatentable over Shaughnessy in view of

Hazama. As to claim 68, Shaughnessy substantially discloses the invention including the claimed

`method of processing symbolic representations of dates stored in a database' as a method and

system for modifying and operating a computer system to perform operations on date fields having a

two digit representation for the year without mistaking the year 2000 and the year 1900 (col.1, lines

7-14 et seq). Shaughnessy also suggests the conversion of all dates within the database from a two

digit format to a four digit format as a viable, but costly alternative for the year 2000 problem (col. 1,

lines 31-46 et seq). The method as recited in the body of the claim is particularly taught by the cited

references as follows:

Shaughnessy discloses the step of providing a database with symbolic representations of dates

stored in at least one date field therein according to a format wherein Y1 Y2 is the numerical year

designator' as a database having a 6 digit-field for storing a Date type in the form of MMDDYY(see

appendix in col. 18, Date Type A), wherein the MM represents the month, DD represents the day and

YY represents the year for a particular six digit-date falling within a 10-decade period of time' as a

date having a cycle or a range of a 100 years (col. 18, Cycle/Range CI = THE DATE CYCLE IS 100

YEARS). As pointed out in column 2, lines 11-14 and column 3, lines 4-8 of Patent No. 5,806,063, all

dates in commercial and industrial databases span within one 100 year. Shaughnessy's system being

of the commercial or industrial kind described in the cited patent, must therefore, as a practical

matter, inherently incorporate this limitation. Shaughnessy also discloses a subroutine for determining

the current date 12 to thereby select a 100 year cycle wherein the current date is the pivot date and

wherein the cycle ends in 100 years from said current date (col.5, lines 31-36). Additionally,

Shaughnessy discloses the step of determining a century designator C1 C2 for each symbolic

representation of a date in the database, CI C2 having a first value if Y1 Y2 is less than YAYB and

having a second value if YIY2 is equal to or greater than YAYB' as the comparison of the current date

to the date when the system was installed with the modifications (modified system install date) to

thereby determine the 32 The current date, by virtue of being the pivot date in the 100 year-cycle and

by being initially set to the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq),

is therefore not any later century value (col.5, lines 36-65 et seq). Alternatively, Shaughnessy

discloses the comparison of the YYMMDD portion 13 of the date to the corresponding date portion at

the end of the 100 year cycle to thereby determine the century value (col.7, lines 7-15 et seq).

Finally, Shaughnessy discloses the step of `reformatting the symbolic representation of each

symbolic representation of a date in at least one date field in the database, without the

addition of any new data field to the database, with the reformatted symbolic representation

of each date in the database having the values C 1 C2, Y 1 Y2, in order to facilitate further

processing of the reformatted symbolic representations of each of the symbolic

representations of each of the dates, by running a program on the reformatted symbolic

representations of each of the dates' by appending the determined century value before the

YYMMDD date in order to yield a CCYYMMDD date format, whereby said appending was

performed without adding a new field to the database (col.5, lines 46-51; col.6, lines 57-65

et seq). Shaughnessy does not particularly detail that the YaYb value for the first decade of the

window is no later than the earliest Y1 Y2 year designator in the database in the selection of

the 10 decade window. However, Hazama discloses an analogous date processing system

having two digit dates spanning from the 20th century to the 21st century stored in memory

(page 2, claim 1) wherein, for a 100 year window (restricting the all the dates in the database

between the 20th and 21st centuries), the pivot date for the window is selected based on a

than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date

which compares low to all other dates (col. 7, lines 16-17 et se44).

two digit year date that is smaller than the smallest two digit year date in the database (e.g. if smallest

two digit date stored in database is 73, the pivot date for the 100 year window is chosen to be 72).

(page 4 of translated document, last paragraph). It would have been obvious to one of ordinary skill in

the art of data processing to combine the teachings of the cited references. The ordinary skilled

artisan having read Shaughnessy would immediately see the need to determine which 100 year span

to use. This determination would have led the

ordinary skilled artisan to the Hazama reference, which teaches the pivot date being smaller.

than the smallest two digit date in the database having all the dates within a 100 year period as a

solution to restrict the selection of Shaughnessy's window and thereby forcing all dates already stored

in the database to fall in the 20th century.

28.   Claim 69 is rejected under 35 U.S.C. . 103(a) as being unpatentable over Shaughnessy in view

of Hazama.

As to claim 69, Shaughnessy substantially discloses the invention including the claimed

method of processing dates in a database' as a method and system for modifying and operating a

computer system to perform operations on date fields having a two digit representation for the year

without mistaking the year 2000 and the year 1900 (col.1, lines 714 et seq). Shaughnessy also

suggests the conversion of all dates within the database from a two digit format to a four digit format

as a viable, but costly alternative for the year 2000 problem (col. 1, lines 31-46 et seq). The method

as recited in the body of the claim is particularly taught by the cited references as follows:

Shaughnessy discloses the step of 'providing a database with dates stored in at least one date field

therein according to a format wherein Y1 Y2 is the numerical year designator' as a database having a

6 digit-field for storing a Date type in the form of MMDDYY(see appendix in cot. 18, Date Type A),

wherein the MM represents the month, DD represents the day and YY represents the year for a

particular six digit-date falling within a 10-decade period of time' as a date having a cycle or a range

of a 100 years (cot. 18, Cycle/Range C I = THE DATE CYCLE IS 100 YEARS). As pointed out in

column 2, lines 11-14 and column 3, lines 4-8 of Patent No. 5,806,063, all dates in commercial and

industrial databases span within one 100 year. Shaughnessy's system being of the commercial or

industrial kind described in the cited patent, must therefore, as a practical matter, inherently

incorporate this limitation. Shaughnessy also discloses a subroutine for determining the current date

14 to thereby select a 100 year cycle wherein the current date is the pivot date and wherein the cycle

ends a 100 years from said current date (col.5, lines 31-36). Additionally, Shaughnessy discloses the

step of 'determining a century designator C1 C2 for each date in the at least one date field of the

database, C1 C2 having a first value if YI Y2 is less than YAYB and having a second value if Y1 Y2 is

equal to or greater than YAYB' as the comparison of the current date to the date when the system

was installed with the modifications (modified system install date) to thereby determine the century

value (co1.5, lines 36-65 et seq). Alternatively, Shaughnessy discloses the comparison of the

YYMM;DD portion 15 of the date to the

The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set to the

operating system date, initially set to 0000 (see cot. 7, lines 19-20 et seq), is therefore not any later

than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date

corresponding date portion at the end of the 100 year cycle to thereby determine the century

value (col.7, lines 7-15 et seq). Finally, Shaughnessy discloses the step of `reformatting the

symbolic representation of each symbolic representation of a date in the at least one date

field in the database, without the addition of any new data field to the database, with the

reformatted symbolic representation of each date in the database having the values C 1 C2,

Y 1 Y2; sorting the reformatted symbolic representations of the dates in the form C 1 C2 Y 1

Y2; and running a program on the reformatted symbolic representations of each of the dates'

by appending the determined century value before the YYMMDD date in order to yield a

CCYYMMDD date format, whereby said appending was performed without adding a new

field to the database (col.5, lines 46-51; co1.6, lines 57-65 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the

window is no later than the earliest YlY2 year designator in the database in the selection of

the 10 decade window. However, Hazama discloses an analogous date processing system

having two digit dates spanning from the 20th century to the 21st century stored in memory

(page 2, claim 1) wherein, for a 100 year window (restricting the all the dates in the database

between the 20th and 21st centuries), the pivot date for the window is selected based on a

two digit year date that is smaller than the smallest two digit year date in the database (e.g. if smallest

two digit date stored in database is 73, the pivot date for the 100 year window is chosen to be 72).

(page 4 of translated document, last paragraph). It would have been obvious to one of ordinary skill in

the art of data processing to combine the teachings of the which compares low to all other dates (col.

7, lines 16-17 et seq).

cited references. The ordinary skilled artisan having read Shaughnessy would immediately see the

need to determine which 100 year span to use. This determination would have led the ordinary skilled

artisan to the Hazama reference, which teaches the pivot date being smaller than the smallest two

digit date in the database having all the dates within a 100 year period as a solution to restrict the

selection of Shaughnessy's window and thereby forcing all dates already stored in the database to fall

in the 20th century.

29.   Claim 70 is rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view

of Hazama, and further in view of Booth.

As to claim 70, Shaughnessy substantially discloses the invention including the claimed

'method for representing and utilizing dates stored in at least one date field of a database utilizing

symbolic representations of the dates stored in at least one date field of the database, which are in a

format that creates ambiguity between dates in each of a pair of adjacent centuries' as a method and

system for modifying and operating a computer system to perform operations on date fields having a

two digit representation for the year without mistaking the year 2000 and the year 1900 (col.1, lines

7-14 et seq). Shaughnessy also suggests the conversion of all dates within the database from a two

digit format to a four digit format as a viable, but costly alternative for the year 2000 problem (col. 1,

lines 31-46 et seq). The method as recited in the body of the claim is particularly taught by the cited

references as follows:

Shaughnessy discloses the steps of 'converting each of the symbolic representations of dates stored

in the at least one date field of the database to a symbolic representation of each

of the respective dates that does not create the ambiguity, by windowing the symbolic

representations of each of the respective dates as stored in the at least one date field of the

database against a pivot year, with the pivot year being less than or equal to the earliest date

represented by the symbolic representation of dates stored in the at least one date field,

without the addition of any new data field to the database, and without modifying any of the

symbolic representations of dates in the at least one date field, for purposes of such

windowing and converting; and running a program on the converted symbolic

representations of each of the dates to manipulate the dates represented by the converted

symbolic representations, separately from the date data symbolic representations contained in

the at least one date field of the database' by providing a database having a 6 digit-field for

storing a Date type in the form of MMDDYY(see appendix in col. 18, Date Type A),

wherein the MM represents the month, DD represents the day and YY represents the year for

a particular six digit-date falling within a 10-decade period of time' as a date having a cycle

or a range of a 100 years (col. 18, Cycle/Range C1 = THE DATE CYCLE IS 100 YEARS).

Shaughnessy further provides a subroutine for determining the current date 36 to thereby select a

100 year cycle wherein the current date is the pivot date and wherein the cycle ends in 100 years

from said current date (col.5, lines 31-36). Additionally, Shaughnessy discloses the comparison of the

current date to the date when the system was installed with the modifications (modified system install

date) to thereby determine the century value (col.5, lines 36-65 et seq). Alternatively, Shaughnessy

provides a subroutine that compares the

The current date, by virtue of being the pivot date in, the 100 year-cycle and by being initially set to

the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any

later YYMMDD portion 17 of the date to the corresponding date portion at the end of the 100 year

cycle to thereby determine the century value (col.7, lines 7-15 et seq). Finally, Shaughnessy

provides a subroutine for appending the determined century value before the YYMMDD date in order

to yield a CCYYMMDD date format, whereby said appending was performed without modifying any of

the date fields stored in the database (col.5, lines 46-51; col.6, lines 57-65 et seq). Shaughnessy

does not particularly detail that the YaYb value for the first decade of the window is no later than the

earliest Y1 Y2 year designator in the database in the selection of the 10 decade window. However,

Hazama discloses an analogous date processing system having two digit dates spanning from the

20th century to the 21st century stored in memory (page 2, claim 1) wherein, for a 100 year window

(restricting the all the dates in the database between the 20th and 21st centuries), the pivot date for

the window is selected based on a two digit year date that is smaller than the smallest two digit year

date in the database (e.g. if smallest two digit date stored in database is 73, the pivot date for the 100

year window is chosen to be 72). (page 4 of translated document, last paragraph). It would have been

obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited

references. The ordinary skilled artisan having read Shaughnessy would immediately see the need to

determine which 100 year span to use. This determination would have led the ordinary skilled artisan

to the Hazama reference, which teaches the pivot date being smaller

than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date, which

compares low to all other dates (col. 7, lines 16-17 et seq.) than the smallest two digit date in the

database having all the dates within a 100 year period as a solution to restrict the selection of

Shaughnessy's window and thereby forcing all dates already stored in the database to fall in the 20th

century.

Shaughnessy and Hazama do not specifically disclose the sorting of converted dates in the form

$C_1 C_2 Y_1 Y_2 M_1 M_2 D_1 D_2$. However, Booth discloses an

analogous system that utilizes the Clipper programming language to process dates stored in a

database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq. In particular, analogously

to Shaughnessy, Booth discloses the SET EPOCH command for comparing an entered two digit date

with the year digit (pivot date) of the epoch setting to determine the century to place the dale into (p.

941, se,,- SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a

corresponding eight digit date (C 1 C2YI Y2M 1 M21) 1132) (see p. 940-941). Additionally, Booth

complements the cited references by suggesting the sorting of converted dates after having been

reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary

skill in the art of data processing at the time of the instant invention to combine the teachings of the

cited references. Booth's teaching of sorted reformatted dates would facilitate the

Shaughnessy-Hazama's system to return the reformatted dates in chronological sequence. Such

disclosed sorting would therefore be very useful when indexing the database in date order, as

suggested by Booth on page 945.

30.   Claim 71 is rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view

of Hazama, and further in view of Booth.

As to claim 71, Shaughnessy substantially discloses the invention including the claimed

'method for representing and utilizing dates stored in at least one date field of the database

utilizing symbolic representations of the dates stored in the at least one date field of the

database, which are in a format that creates ambiguity between dates in each of a pair of

adjacent centuries' as a method and system for modifying and operating a computer system

to perform operations on date fields having a two digit representation for the year without

mistaking the year 2000 and the year 1900 (col. 1, lines 7-14 et seq). Shaughnessy also

suggests the conversion of all dates within the database from a two digit format to a four digit format

as a viable, but costly alternative for the year 2000 problem (col. 1, lines 31-46 et seq). The method

as recited in the body of the claim is particularly taught by the cited references as follows:

Shaughnessy discloses the steps of 'converting each of the symbolic representations of dates stored

in the at least one date field of the database to a symbolic representation of each of the respective

dates that does not create the ambiguity, by windowing the symbolic representations of each of the

respective dates as stored in the at least one date field of the database against a pivot year, with the

pivot year being less than or equal to the earliest date represented by a symbolic representation of

dates stored in the at least one date field, and without the addition of any new data field to the

database for purposes of such windowing and converting; and running a program on the stored

converted symbolic representations of each of the converted symbolic representations of the dates to

manipulate the dates represented by the converted symbolic representations, separately from the

date data symbolic representations contained in the at least one (late field of the database' by

providing a database having a 6 digit-field for storing a Date type in the form of

MMDDYY(see appendix in col. 18, Date Type A), wherein the MM represents the month,

DD represents the day and YY represents the year for a particular six digit-date falling within

a 10-decade period of time' as a date having a cycle or a range of a 100 years (col. 18,

Cycle/Range CI = THE DATE CYCLE IS 100 YEARS). Shaughnessy further provides a

subroutine for determining the current date 38 to thereby select a 100 year cycle wherein the

current date is the pivot date and wherein the cycle ends in 100 years from said current date

(col.5, lines 31-36). Additionally, Shaughnessy discloses the comparison of the current date

to the date when the system was installed with the modifications (modified system install

date) to thereby determine the century value (col.5, lines 36-65 et seq). Alternatively,

Shaughnessy provides a subroutine that compares the YYMMDD portion 39 of the date to the

corresponding date portion at the end of the 100 year cycle to thereby determine the century

value (col.7, lines 7-15 et seq). Finally, Shaughnessy provides a subroutine for appending

the determined century value before the YYMMDD date in order to yield a CCYYMMDD

date format, whereby said appending was performed without modifying any of the date fields stored in

the database (col. 5, lines 46-51; col.6, lines 7-65 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of

"The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set

to the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any

later than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might bi° desirable to set the current date to a date

which compares low to all other dates (col. 7, lines 16-17 et seq).

the window is no later than the earliest YIY2 year designator in the database in the selection of the 10

decade window. However, Hazama discloses an analogous date processing system having two digit

dates spanning from the 20th century to the 21st century stored in memory, (page 2, claim 1)

wherein, for a 100 year window (restricting the all the dates in the database between the 20th and

21st centuries), the pivot date for the window is selected based on a two digit year date that is smaller

than the smallest two digit year date in the database (e.g. if smallest two digit date stored in database

is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated document, last

paragraph). It would have been obvious to one of ordinary skill in the art of data processing to

combine the teachings of the cited references. The ordinary skilled artisan having read Shaughnessy

would immediately see the need to determine which 100 year span to use- This determination would

have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being

smaller than the smallest two digit date in the database having all the dates within a 100 year period

as a solution to restrict the selection of Shaughnessy's window and thereby forcing all dates already

stored in the database to fall in the 20th century.

Shaughnessy and Hazama do not specifically disclose the sorting the converted dates in the form

C.sub. I C.sub.2 Y.sub.I Y.sub.2 M.sub. I M.sub.2 D.sub. I D.sub.2. However, Booth discloses an

analogous system that utilizes the Clipper programming language to process dates stored in a

database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq. In particular, analogously

to Shaughnessy, Booth discloses the SET EPOCH command for comparing an entered two digit date

with the year digit (pivot date) of the epoch setting to determine the century to place the date into (p.

941, see SET EPOCH paragraph et seq) thereby converting a six digit date (MIV1DDYY) into a

corresponding eight digit date (CIC2YIY2MIM2DID2) (see p. 940-941). Additionally, Booth

complements Shaughnessy by suggesting the sorting of converted dates after having; been

reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary

skill in the art of data processing at the time of the instant invention to combine the teachings of the

cited references. Booth's teaching of sorted reformatted dates would facilitate the Shaughnessy-

Hazama's system to return the reformatted dates in chronological sequence. Such disclosed sorting

would therefore be very useful when indexing t:2e database in date order, as suggested by Booth on

page 945.

Shaughnessy and Hazama do not specifically disclose the step of `storing the converted symbolic

representations separate from the at least one date field in the database.' Booth, however, further

complements the cited references by renaming and storing sorted dates in the CUSTMER.DBF NEW

CUST.DBF databases (p. S41). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. Booth's teaching of the storing the

sorted reformatted dates would users of the Shaughnessy-Hazama's system to readily retrieve the

reformatted dates in their chronological sequence at any time and in accordance with their needs.

31.   Claims 72 is rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view

of Hazama.

       As to claim 72, Shaughnessy substantially discloses the invention including the claimed

Application/Control Number ●9/512,592; 90/005,592; 90/005,62●0/005727;   Page
90/006,541   94
Art Unit: 2161

'method of processing symbolic representations of dates stored in a database' as a method and system for modifying and operating a computer system to perform operations on date fields having a two digit representation for the year without mistaking the year 2000 and the year 1900 (col. 1, lines 7-14 et seq). Shaughnessy also suggests the conversion of all dates within the database from a two digit format to a four digit format as a viable, but costly alternative for the year 2000 problem (col. 1, lines 31-46 et seq). The method as recited in the body of the claim is particularly taught by the cited references as follows: Shaughnessy discloses the step of 'selecting a database with symbolic representations of dates stored therein according to a format wherein MI 142 is the numerical month designator, $D_1 D_2$ is the numerical day designator, and $Y_1 Y_2$ is the numerical year designator' as a database having a 6 digit-field for storing a Date type in the form of MMDDYY(see appendix in col. 18, Date Type A), wherein the MM represents the month, DD represents the day and YY represents the year for a particular six digit-date falling within a 10-decade period of time' as a date having a cycle or a range of a 100 years (col. 18, Cycle/Range C_1 = THE DATE CYCLE IS 100 YEARS). As pointed out in column 2, lines 11-14 and column 3, lines 4-8 of Patent No. 5,806,063, all dates in commercial and industrial databases span within one 100 year. Shaughnessy's system being of the commercial or industrial kind described in the cited patent, must therefore, as a practical matter, inherently incorporate this limitation. Shaughnessy also a subroutine for determining the current date 40 to thereby select a 100 year cycle wherein the current date is the pivot date and wherein the cycle ends a 100 The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set to the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any later years from said current date (col.5, lines 31-36). Additionally, Shaughnessy discloses the step of' determining a century designator $C_1 C_2$ for each symbolic representation of a date in the database, $C_1 C_2$ having a first value if $Y_1 Y_2$ is less than YAYB and having second value if

Y1 Y2 is equal to or greater than YAYB' as the comparison of the current date to the date when the

system was installed with the modifications (modified system install date) to thereby determine the

century value (col.5, lines 36-65 et seq). Alternatively, Shaughnessy discloses the comparison of the

YYMMDD portion 41 o F the date to the corresponding date portion at the end of the 100 year cycle

to thereby determine the century value (col.7, lines 715 et seq). Finally, Shaughnessy discloses the

step of 'reformatting the symbolic representation of each symbolic representation of a date in the

database with the values CI C2, YI Y2, M1 M2, and DI D2 prior to collectively further processing

information contained within the database associated with the respective dates' by appending the

determined century value before the YYMMDD (late in order to yield a CCYYMMDD date format,

whereby said appending was performed without adding a new field to the database (col.5, lines

46-51; col.6, lines 57-65 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest YIY2 year designator in the database in the selection of the 10 decade window.

However, Hazama discloses am analogous date processing system having two digit dates spanning

from the 20th century to the 21st century stored in memory than the earliest date in 100 year-cycle in

the database.

Shaughnessy specifically suggests that it might b.- desirable to set the current date to a date

which compares low to all other dates (col. 7, lines 16-17 et seq).

(page 2, claim 1) wherein, for a 100 year window (restricting the all the dates in the database

between the 20th and 21st centuries), the pivot date for the window is selected based on a two digit

year date that is smaller than the smallest two digit year date in the database (e.g. if smallest two digit

date stored in database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of

translated document, last p,3ragraph). It would have been obvious to one of ordinary skill in the art of

data processing to combine the teachings of the cited references. The ordinary skilled artisan having

read Shaughnessy would immediately see the need to determine which 100 year span to use. This

determination would have led the ordinary skilled artisan to the Hazama reference, which teaches the

pivot date being smaller than the smallest two digit date in the database having all the dates within a

100 year period as a solution to restrict the selection of Shaughnessy's window and thereby forcing all

dates already stored in the database to fall in the 20th century.

32.   Claims 73 is rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view

of Hazama.

As to claim 73, Shaughnessy substantially discloses the invention including the claimed

'method of processing symbolic representations of dates stored in a database' as a method and

system for modifying and operating a computer system to perform operations on date fields having a

two digit representation for the year without mistaking the year 2000 and the year 1900 (col.1, lines

7-14 et seq). Shaughnessy also suggests the conversion of all dates within the database from a two

digit format to a four digit format as a viable, but costly alternative for the year 2000 problem (col. 1,

lines 31-4.6 et seq). The method as recited in the body of the claim is particularly taught by the cited

references as follows: Shaughnessy discloses the step of 'providing a database with symbolic

representations of dates stored therein according to a format wherein YIY2 is the numerical year

designator, all of the symbolic representations of dates falling within a 10-decade period of time' as a

database having a 6 digit-field for storing a Date type in the form of MMDDYY(see appendix in col.

18, Date Type A), wherein the MM represents the month, DD represents the day and YY represents

the year for a particular six digit-date falling within a 10-decade period of time' as a date having a

cycle or a range of a 100 years (col. 18, Cycle/Range C 1 THE DATE CYCLE IS 100 YEARS). As

pointed out in column 2, lines 11-14 and column 3, lines 4-8 of Patent No. 5,806,063, all dates in

commercial and industrial databases span within one 100 year. Shaughnessy's system being of the

commercial or industrial kind described in the cited patent, must therefore, as a practical matter,

inherently incorporate this limitation. Shaughnessy also discloses a subroutine for determining the

current date 42 to thereby select a 100 year cycle wherein the current date is the pivot date and

wherein the cycle ends a 100 years from said current date (col.5, lilies 31-36). Additionally,

Shaughnessy discloses the step of 'determining a century designator C1 C2 for each symbolic

representation of a date in the database, CI C2 having a first value if Y1 Y2 is less than YAYB and

having a second value if Y 1 Y2 equal to or greater than YAYB' as the comparison of the current date

to the date when the system was installed with the

The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially

set to the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not

any late

modifications (modified system install date) to thereby determine the century value (col.5, lines 36-65

et seq). Alternatively, Shaughnessy discloses the comparison of the YYMMDD portion 43 of the date

to the corresponding date portion at the end of the 100 year cycle to thereby determine the century

value (col.7, lines 7-15 et, seq). Finally, Shaughnessy discloses the step of 'reformatting the symbolic

representation of the date with the values CI C2, YI Y2, to facilitate further processing of the dates' by

appending the determined century value before the YYMMDD date in order to yield a CCYYMMDD

date format, whereby said appending was performed without adding a new field to the database

(col.5, lines 46-51; co1.6, lines 57-65 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest YlY2 year designator in the database in the selection of the 10 decade window.

However, Hazama discloses an analogous date processing system having two digit dates spanning

from the 20th century 1:o the 21st century stored in memory (page 2, claim 1) wherein, for a 100 year

window (restricting the all the dates in the database between the 20th and 21st centuries), the pivot

date for the window is selected based on a two digit year date that is smaller than the smallest two

digit year date in the database (e.g. if smallest two digit date stored in database is 73, the pivot date

for the 100 year window is chosen to be 72). (page 4 of translated document, last paragraph). It

would have been obvious to one of ordinary skill in the art of data processing to combine the

teachings of the than the earliest date in 100 year-cycle in the database.

Shaughnessy specifically suggests that it might be desirable to set the current date to a date, which

compares low to all other dates (col. 7, lines 16-17 et seq).

cited references. The ordinary skilled artisan having read Shaughnessy would immediately see the

need to determine which 100 year span to use. This determination would have led the ordinary skilled

artisan to the Hazama reference, which teaches the pivot date being smaller than the smallest two

digit date in the database having all the dates within a 100 year period as a solution to restrict the

selection of Shaughnessy's window and thereby forcing all dates already stored in the database to fall

in the 20th century.

33. Claim 74 is rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view of

Hazama, and further in view of Booth.

As to claim 74, Shaughnessy substantially discloses the invention including the claimed

'method of processing dates in a database' as a method and system for modifying and operating a

computer system to perform operations on date fields having a two digit representation for the year

without mistaking the year :'000 and the year 1900 (col.1, lines 714 et seq). In particular,

Shaughnessy discloses the step of 'providing a database with symbolic representations of dates

stored therein according to a format wherein Y1 Y2 is the numerical year designator, all of symbolic

representations of dates falling within a 10-decade period of time' as a database having a 6 digit-field

for ;;toning a Date type in the form of MMDDYY(see appendix in col. 18, Date Type A), wherein the

MM represents the month, DD represents the day and YY represents the year for a particular six

digit-date falling within a 10-decade period of time' as a date having a cycle or a range of a 100 years

(col. 18, Cycle/Range C1 = THE DATE CYCLE IS 100 YEARS). As pointed out in column 2, lines

11-14 and column 3, lines 4-8 of Patent No. 5,806,06?, all dates in commercial and

industrial databases span within one 100 year. Shaughnessy's system being of the

commercial or industrial kind described in the cited patent, must therefore, as a practical

matter, inherently incorporate this limitation. Shaughnessy also discloses the claimed step of

'selecting a 10-decade window with a YAYB value for the first decade of the window, YAYB being no

later than the earliest Y 1 Y2 year designator in the database' as a subroutine for determining the

current date 44 to thereby select a 1C0 year cycle wherein the current date is the pivot date and

wherein the cycle ends in 100 years from said current date (col.5, lines 31-36). Additionally,

Shaughnessy discloses the step of 'determining a century designator CI C2 for each date in the

database, C1 C2 having a first value if Y1 Y2 is less than YAYB and having a second value if Y1 Y2

is equal to or greater than YAYB' as the comparison of the current date to the date when the system

was installed with the modifications (modified system install date) to thereby determine the century

value (col.5, lines 36-65 et seq). Alternatively, Shaughnessy discloses the comparison of the

YYMMDD portion 41 of the date to the corresponding date portion at the end of the 100 year cycle to

thereby determine the century value (col.7, lines 7-15 et seq). Finally, Shaughnessy discloses the

step of 'reformatting each date in the form CIC2YlY2 to facilitate further processing of the dates' by

appending the determined century value before the YYMMDD date in order to yield a

"The current date, by virtue of being the pivot date in the 100 year-cycle and by being initially set to

the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any

later than the earliest date in 100 year-cycle in the database.

as Shaughnessy specifically suggests that it might b - desirable to set the current date to a date,

which compares low to all other dates (col. 7, lines 16-17 et seq).

CCYYMMDD date format, whereby said appending was performed without adding a new

field to the database (col.5, lines 46-51; co1.6, lines 57–65 et seq).

In the event that applicant were to argue that Shaughnessy does not disclose the limitation

that the two digit date is smaller or equal to the smallest date in the database, where all the

dates in the database fall within a 100 year period, it would have been obvious to the ordinary skilled

artisan to look to the teachings of Hazama to complement Shaughnessy's.

Hazama discloses an analogous date processing system wherein, for a 100 year window (restricting

the all the dates in the database between the 20th and 21st centuries), the pivot date for the window

is selected based on a two digit year date that is smaller than the smallest two digit year date in the

database (e.g. if smallest two digit date stored in database is 73, the pivot date for the 100 year

window is chosen to be 72). (page 4 of translated document, last paragraph). It would have been

obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited

references. The ordinary skilled artisan having read Shaughnessy would immediately see the need to

determine which 100 year- span to use. This determination would have led the ordinary skilled artisan

to the Hazama reference, which teaches the pivot date being smaller than the smallest two digit date

in the database having all the dates within a 100 year period as a solution to restrict the selection of

Shaughnessy's window and thereby forcing all dates already stored in the database to fall in the 20th

century.

Shaughnessy and Hazama do not specifically disclose the sorting the converted dates in the form

$C_{.sub.1} C_{.sub.2} Y_{sub.1} Y_{.sub.2} M_{.sub.1} M_{.sub.2} D_{.sub.1} D_{.sub.2}$. However, Booth discloses an

analogous system that utilizes the Clipper programming language to process dates stored in a

database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq. In particular, analogously

to Shaughnessy, Booth discloses the SET EPOCH command for comparing an entered two digit date

with the year digit (pivot date) of the epoch setting to determine the century to place the date into (p.

941, see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a

corresponding eight digit date (C1C2YIY2MIM2DID2) (seep. 940-941). Additionally, Booth

complements Shaughnessy and Hazama by suggesting the sorting of converted dates after having

been reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of

ordinary skill in the art of data processing at the time. of the instant invention to combine the

teachings of the cited references. Booth's teaching of sorted reformatted dates would facilitate the

Shaughnessy-Hazama's system to return the reformatted dates in chronological sequence. Such

disclosed sorting would therefore be very useful when indexing the database in date order, as

suggested by Booth on page 945.

34.   Claims 75 is rejected under 35 U.S.C. 103(a) as obvious over Shaughnessy in view of

Hazama.

As to claim 75, Shaughnessy substantially discloses the invention including the claimed

`method of processing symbolic representations of dates stored in a database' as a method and

system for modifying and operating a computer system to perform operations on date fields having a

two digit representation for the year without mistaking the year 2000 and the year 1900 (col. 1, lines

7-14 et seq). Shaughnessy also suggests the conversion of all dates within the database from a two

digit format to a four digit format as a viable, but costly alternative for the year 2000 problem (col. 1,

lines 31-46 et seq). The method as recited in the body of the claim is particularly taught by the cited

references as follows:

Shaughnessy discloses the step of 'providing a database with symbolic representations of dates

stored therein according to a format wherein MIM2 is the numerical month designator, DI D2 is the

numerical day designator, and YI Y2 is the numerical year designator' as a database having a 6

digit-field for storing a Date type in the form of MMDDYY(see appendix in col. 18, Date Type A),

wherein the MM represents the month, DD represents the day and YY represents the year for a

particular six digit-date falling within a 10-decade period of time' as a date having a cycle or a range

of a 100 years (col. 18, Cycle/Range C1 = THE DATE CYCLE IS 100 YEARS). As pointed out in

column 2, lines 11-14 and column 3, lines 4-8 of Patent No. 5,806,063, all dates in commercial and

industrial databases span within one 100 year. Shaughnessy's system being of the commercial or

industrial kind described in the cited patent, must therefore, as a practical matter, inherently

incorporate this limitation. Shaughnessy also discloses a subroutine for determining the current date

to thereby select a 100-year cycle wherein the current date is the pivot date and wherein the cycle

ends in 100 years from said current date (co1.5, lines 31-36). Additionally, Shaughnessy discloses

the step of determining a century designator CIC2 for each symbolic representation of a date in the

database, C1 C2 having ,a first value if Y1 Y2 is less than YAYB and having a second value if Y1 Y2

is equal to or greater than YAYB' as the comparison of the current date to the date when the system

was installed with the ".he current date, by virtue of being the pivot date in the 100 year-cycle and by

being initially set to the operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is

therefore not any later modifications (modified system install date) to thereby determine the century

value (co1.5, lines 36-65 et seq). Alternatively, Shaughnessy discloses the comparison of the

YYMMDD portion 47 of the date to the corresponding date portion at the end of the 100 year cycle to

thereby determine the century value (col.7, lines 7-15 et seq). Finally, Shaughnessy discloses the

step of 'reformatting the symbolic representation of each symbolic representation of a date in the

database, without the addition of any new data field to the database, with the reformatted symbolic

representation of each date in the database having the values C 1 C2, YIY2, MIM2, and D1 D2 in

order to facilitate further processing of the reformatted symbolic representations of each of the

symbolic representations of each of the dates' by appending the determined century value before the

YYMMDD date in order to yield a CCYYMMDD date format, whereby said appending was performed

without adding a new field to the database (col.5, lines 46-51; col.6, lines 57-65 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest YIY2 year designator in the database in the selection of the 10 decade window.

However, Hazama discloses an analogous date processing system having two digit dates spanning

from the 20th century to the 21st century stored in memory (page 2, claim 1) wherein, for a 100 year

window (restricting the all the dates in the database between the 20th and 21st centuries), the pivot

date for the window is selected based on a two digit year date that is smaller than the smallest tyro

digit year date in the database (e.g. if than the earliest date in 100 year-cycle in the database.

4' Shaughnessy specifically suggests that it might be desirable to set the current date to a date

which compares low to all other dates (col. 7, lines 16-17 et seq). smallest two digit date stored in

database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated

document, last paragraph). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. The ordinary skilled artisan having read

Shaughnessy would immediately see the need to determine which 100 year span to use. This

determination would have led the ordinary skilled artisan to the Hazama reference, which teaches the

pivot date being smaller than the smallest two digit date in the database having all the dates within a

100 year period as a solution to restrict the selection of Shaughnessy's window and thereby forcing all

dates already stored in the database to fall in the 20th century.

35.   Claims 76 is rejected under 35 U.S.C. 103(a) as being unpatentable over Shaughnessy in view

of Hazama, and further in view of Booth.

As to claim 76, Shaughnessy substantially discloses the invention including the claimed

'method of processing dates in a database' as a method and system for modifying and operating a

computer system to perform operations on date fields having a two digit representation for the year

without mistaking the year 2000 and the year 1900 (col.1, lines 714 et seq). Shaughnessy also

suggests the conversion of all dates within the database from a two digit format to a four digit format

as a viable, but costly alternative for the year 2000 problem (col. 1, lines 31-46 et seq). The method

as recited in the body of the claim is particularly taught by the cited references as follows:

Shaughnessy discloses the step of providing a database with dates stored therein according to a

format wherein M1 M2 is the numerical month designator, DI D2 is the numerical day designator, and

YI Y2 is the numerical year designator' as a database having a 6 digit-field for storing a Date type in

the form of MMDDYY(see appendix in col. 18, Date Type A), wherein the MM represents the month,

DD represents the day and YY represents the year for a particular six digit-date falling within a

10-decade period of time as a date having a cycle or a range of a 100 years (col. 18, Cycle/Range C1

THE DATE CYCLE IS 100 YEARS). As pointed out in column 2, lines 11-14 and column 3, lines 4-8

of Patent No. 5,806,063, all dates in commercial and industrial databases span within one 100 year.

Shaughnessy's system being of the commercial or industrial kind described in the cited patent, must

therefore, as a practical matter, inherently incorporate this limitation. Shaughnessy also discloses a

subroutine for determining; the current date 48 to thereby select a 100 year cycle wherein the current

date is the pivot date and wherein the cycle ends a 100 years from said current date (col.5, lines

31-36). Additionally, Shaughnessy discloses the step of 'determining a century designator C 1 C2 for

each date in the database, C 1 C2 having a first value if Y 1 Y2 is less than YAYB and having a

second value if Y 1 Y2 is equal to or greater than YAYB' as the comparison of the current date to the

date when the system was installed with the modifications (modified system install date) to thereby

determine the century value (col.5, lines 36-65 et seq). Alternatively, Shaughnessy discloses the

The current date, by virtue of being the pivot date vi the 100 year-cycle and by being initially set to the

operating system date, initially set to 0000 (see col. 7, lines 19-20 et seq), is therefore not any later

than the earliest date in 100 year-cycle in the database.

comparison of the YYMMDD poirtion49 of the date to the corresponding date portion at the end of the

100 year cycle to thereby determine the century value (co1.7, lines 7-15 et seq). Finally,

Shaughnessy discloses the step of `reformatting the symbolic representation of each symbolic

representation of a date in the database, with the addition of any new data field to the database, with

the reformatted symbolic representation of each date in the database having the values C1 C2, Y1

Y2, MI M2, and DI D2, in order to facilitate further processing of the reformatted symbolic

representations of each of the symbolic representations of each of the dates' by appending the

determined century value before the YYMMDD date in order to yield a CCYYMMDD date format,

whereby said appending was performed without adding a new field to the database (col.5, lines

46-51; col.6, lines 57-65 et seq).

Shaughnessy does not particularly detail that the YaYb value for the first decade of the window is no

later than the earliest YlY2 year designator in the database in the selection of the 10 decade window.

However, Hazama discloses an analogous date processing system having two digit dates spanning

from the 20th century to the 21st century stored in memory (page 2, claim 1) wherein, for a 100 year

window (restricting the all the dates in the database between the 20th and 21st centuries), the pivot

date for the window is selected based on a two digit year date that is smaller than the smallest two

digit year date in the database (e.g., if smallest two digit date stored in database is 73, the pivot date

for the 100 year window is chosen to be 72). (page 4 of translated document, last paragraph). It

would have been obvious to one of ordinary skill in the art of data processing to combine the

teachings of the Shaughnessy specifically suggests that it might be desirable to set the current date

to a date which compares low to all other dates (col. 7, lines 16-17 et seq).

cited references. The ordinary skilled artisan having read Shaughnessy would immediately see the

need to determine which 100 year span to use. This determination would have led the ordinary skilled

artisan to the Hazama reference, which teaches the pivot date being smaller than the smallest two

digit date in the database having all the dates within a 100 year- period as a solution to restrict the

selection of Shaughnessy's window and thereby forcing all dates already stored in the database to fall

in the 20th century.

Shaughnessy and Hazama do not specifically disclose the sorting the converted dates in the form

C.sub. l C.sub.2 Y.sub. l Y.sub.2 M.sub. l M.sub.2 D.sub. l D.sub.2. However, Booth discloses an

Application/Control Number ● 9/512,592; 90/005,592; 90/005,62 ● 0/005727;   Page
90/006,541   107
Art Unit: 2161

analogous system that utilizes the Clipper programming language to process dates stored in a database to thereby derive other dates therefrom (p. 939, lines 1-3 et seq). In particular, analogously to Shaughnessy, Booth discloses the SET EPOCH command for comparing an entered two digit date with the year digit (pivot date) of the epoch setting to determine the century to place the date into (p. 941, see SET EPOCH paragraph et seq) thereby converting a six digit date (M:VIDDYY) into a corresponding eight digit date (CIC2YIY2MIM2DID2) (see p. 940-941). Additionally, Booth complements Shaughnessy and Hazama by suggesting the sorting of converted dates after having been reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary skill in the art of data processing at the time of the instant invention to combine the teachings of the cited references. Booth's teaching of sorted reformatted dates would facilitate Shaughnessy-Hazama's system to return the reformatted dates in chronological sequence. Such disclosed sorting would therefore be very useful when indexing the database in date order, as suggested by Booth on page 945.

36. Claims 1-3, 5, 7, 9-10 are rejected under 35 U.S. C. 103(a) as being unpatentable over B.G. Ohms, Computer processing of-Dates Outside the Twentieth Century, IBM Systems Journal, Volume 25, Number 2, 1986, pages 244-251, (Ohms, hereinafter), in view of Hazama.

As to claim 1, Ohms substantially discloses the invention including the claimed 'method of processing symbolic representations of dates stored in a database' by presenting a computer implemented method for processing date outside the twentieth century (see title, p 244 et seq). In particular, Ohms discloses the step of 'providing a database with symbolic representations of dates stored therein according to a format wherein $M_1 M_2$ is the numerical month designator,

Application/Control Number ●9/512,592; 90/005,592; 90/005,62●0/005727;   Page
90/006,541   108
Art Unit: 2161

$D_1 D_2$ is the numerical day designator, and $Y_1 Y_2$ is the numerical year

designator, all of the symbolic representations of dates falling within a 10-decade period of time' by

detailing a short Gregorian format (MMDDYY) to represent dates (p 247, see table 1), wherein said

dates fall within a 10 decade or a 100 year window (p 249, left-hand column, lines 3-7 et seq). Ohms

also discloses the step of 'determining a century designator $C_1 C_2$ for each symbolic

representation of a date in the database, $C_1 C_2$ having a first value if ~$Y_1 Y_2$ is

less than $Y_A Y_B$ and having a second value if $Y_1 Y_2$ is equal to or greater than

$Y_A Y_B$' by indicating that years that are later or equal (25-99) to the pivot date (25) would

fall within the 20th century thereby equating C 1 C2 to 19 (i. e. 1925-1999), whereas dates that are

earlier 900-24) than the pivot date would fall within the 21st century thereby equating CIC2 to 20 (i.e.

2000-2024) (see p 248, right-hand column. Finally, OHMS discloses the step of 'reformatting the

symbolic representation of the date with the values $C_1 C_2$, $Y_1 Y_2$, M_1

M_2, and $D_1 D_2$ to facilitate further processing of the dates' by indicating that upon

determining that a two-digit date falls within the 20th or the 21st the century50, it is expressed in

accordance with its corresponding century (i.e. 25-99 ----> 1925-1999, and 00-24 ----> 2000-2024) (p

248, right hand column).

Regarding the step of 'selecting a 10-decade window with a $Y_A Y_B$ value for the first

decade of the window, Y.SubA Y.sub.B being no later than the earliest $Y_1 Y_2$ year

designator in the database,' Ohms discloses specifying a year as the desired starting point (pivot

date) of the 100 year window (p 248, right hand column, 2nd paragraph). Ohms does not particularly

detail that the pivot date is earlier than the earliest two digit date in the database. Hazama, however,

discloses an analogous date processing system wherein, for a 100 year window, the pivot date for the

window is selected based on a two digit year date that is smaller than the smallest two digit year date

in the database (e.g. if smallest two digit date stored in database is 73, the pivot date for the 100 year

window is chosen to be 72). (page 4 of translated document, last paragraph). It would have been

obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited

references. The ordinary skilled artisan having read Ohms would immediately see the need to

determine which 100 year span to use. This determination would have led the ordinary skilled artisan

to, (soO1uns implicitly discloses that CIC2 corresponds to 19 or 20 depending on whether the date is

less than or greater than or equal to the pivot date), the Hazama reference, which teaches the pivot

date being smaller than the smallest two digit date in the database having all the dates within a 100

year period as a solution to restrict the selection of Ohms's window and thereby forcing all dates

already stored in the database to fall in the 20th century.

As to claim 2, Ohms and Hazama disclose the invention as discussed in the rejection of claim 1.

Additionally, Ohms discloses that `the 10-decade window includes the decade beginning in the year

2000' by indicating that the 100 year window contains dates that span in the 21st century (2000-2024)

(p 248, right hand column).

As to claim 3, Ohms and Hazama disclose the invention as discussed in the rejection of claim 2.

Additionally, Ohms discloses that the step of `determining includes the step of determining the first

value as 20 and the second value ~Ls 19' by indicating that dates that are greater or equal to the

pivot date fall within the 20th century (CIC2=19) and dates that are less than the pivot date fall within

the 21st century (C1 C2=20) p. 2488, right hand column).

As to claim 5, Ohms and Hazama disclose the invention as discussed in the rejection of claim 1.

Additionally, Ohms discloses that the step of `reformatting includes the step of reformatting each

symbolic representation of a date into the format $C.sub.1 C.sub.2 Y.sub.1 Y.sub.2 M.sub.1 M.sub.2$

$D.sub.1 D.sub.2$ by indicating that dates that fall within the 20th century (greater than or equal to the

pivot date) are preceded by 19 (e.g. 1925-1999), whereas dates that fall within the 21st century (less

than the pivot date) are preceded by 20 (e.g. 20002024) p 2477, right hand column).

As to claim 7, Ohms and Hazama disclose the invention as discussed in the rejection of claim 1.

Additionally, Ohms discloses that the step of 'providing a database includes the step of converting

pre-existing date information having a different format into the format wherein $M.sub.1 M.sub.2$ is the

numerical month designator, $D sub.1 D.sub.2$ is the numerical day designator and $Y.sub.1 Y.sub.2$ is

the numerical year designator' by suggesting the conversion from of a date from a Gregorian format

(M]VIDDYYYY) to a short Gregorian format (YYMMDD), wherein YY indicates the year, MM indicates

the month and DD indicates the date p 2477, table 1).

As to claim 9, Ohms and Hazama disclose the invention as discussed in the rejection of claim 1.

Additionally, Ohms discloses, 'after the step of reformatting, the storing of the symbolic representation

of dates and their associated information back into the database' by suggesting that converted eight

dates are stored in the database although they take up more memory space than non-converted six

digits dates p 2499, left hand column).

As to claim 10, Ohms and Hazama disclose the invention as discussed in the rejection of claim 9.

Additionally, Ohms discloses, 'after the step of reformatting, the manipulating of information in the

Application/Control Number● ●/512,592; 90/005,592; 90/005,628● ●0/005727;   Page
90/006,541   111
Art Unit: 2161

database having the reformatted date information therein' by suggesting that the converted dates can be saved in the database p 2499, left hand column).

37.   Claims 4, 6, 8 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of Hazama, as applied to the rejection of claims 1-3, 5, 7, 9-10 above, further in view of Booth.

As to claim 4, Ohms and Hazama substantially disclose the invention as discussed in the rejection of claim 1 above. Ohms and Hazama do not particularly disclose the additional step of `sorting the symbolic representations of dates, after the step of reformatting.' However, Booth discloses an analogous system that utilizes the Clipper programming language to process dates stored in a database to thereby derive other dates therefrom p 939, lines 1-3 et seq. In particular, analogously to Ohms, Booth discloses the SET EPOCH command for comparing an entered two digit date with the year digit (pivot date) of the epoch setting to determine the century to place the date into p 9411, see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a corresponding eight digit date (CIC2YIY2MIM2DID2) (see p. 940-941). Additionally, Booth complements Ohms and Hazama by suggesting the sorting of converted dates after having been reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary skill in the art of data processing at the time of the instant invention to combine the teachings of the cited references. Booth's teaching of sorted reformatted dates would facilitate the Ohms-Hazama's system to return the reformatted dates in chronological sequence. And, it would therefore be very useful when indexing the database in date order, as suggested by Booth in page 945.

As to claim 6, Ohms and Hazama substantially disclose the invention as discussed in the rejection of

claim 5 above. Ohms and Hazama do not specifically, disclose the additional

step of sorting the symbolic representations of dates using a numerical-order sort, after the step of

reformatting.' However, Booth discloses an analogous system that utilizes the Clipper programming

language to process dates stored in a database to thereby derive other dates therefrom (p 939, lines

1-3 et seq. In particular, analogously to Ohms, Booth discloses the SET EPOCH command for

comparing an entered two digit date with the year digit (pivot date) of the epoch setting to determine

the century to place the date into (p 941, see SET EPOCH paragraph et seq) thereby converting a six

digit- date (MMDDYY) into a corresponding eight digit date (C I C2Y I Y2M 1 M2D 1 E,2) (see p.

940-941). Additionally, Booth complements Ohms and Hazama by suggesting the sorting of

converted dates after having been reformatted by the SET EPOCH command (p. 945). It would have

been obvious to one of ordinary skill in the art of data processing at the time of the instant invention to

combine the teachings of the cited references. Booth's t.-aching of sorted reformatted dates would

facilitate the Ohms-Hazama's system to return the reformatted dates in chronological sequence. And,

it would therefore be very useful when indexing the database in date order, as suggested by Booth in

page 945.

As to claim 8, Ohms and Hazama substantially disclose the invention as discussed in the rejection of

claim 1. Ohms and Hazama do not specifically, disclose the step of selecting Y.sub.A Y.sub.B such

that Y.Sub.B is 0 (zero). However, Booth discloses an analogous system that utilizes the Clipper

programming language to process dates stored in a database to thereby derive other dates therefrom

(p 939, lines 1-3 et seq.) In particular, analogously to Ohms and Hazama, Booth discloses the SET

EPOCH command for comparing an entered two digit date with the year digit (pivot date) of the epoch

setting to determine the century to place the date into (p 941, see SET EPOCH paragraph et seq)

thereby converting a six digit date (MMDDYY) into a corresponding eight digit date; (C1 C2Y 1 Y2M I

M2D 1132) (see p. 940-941). Additionally, Booth complements Ohms and Hazama by suggesting that

the pivot date be set to 90 by selecting set epoch to be 1990 (i.e. YAYB = 90), such that YB equals to

zero (p. 942). It would have been obvious to one of ordinary skill in the art of data processing at the

time of the instant invention to combine the teachings of the cited references. Booth's teaching of

setting the pivot date to a predetermined value would enable users of the Ohms-Hazama's system to

have direct control over the return of reformatted dates to thereby preset the date processing system

in accordance with their needs.

38. Claims 11-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 11, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed 'method of processing dates in a database' by presenting a computer implemented method

for processing date outside the twentieth century (see title, p 244 et seq). Ohms further discloses the

step of 'providing a database with dates stored therein according to a format wherein M.sub. I M.sub.2

is the numerical month designator, D.sub. ID.sub.2 is the numerical day designator, and Y.sub. I

Y.sub.2 is the numerical year designator, all of dates falling within a 10-decade period of time which

includes the decade beginning in the year 2000' by detailing a short Gregorian format (MMDDYY) to

represent dates (p 247, see table 1), wherein said dates fall within a 10 decade or a 100 year window

(p 249, left-hand column, lines 3-7 et seq) and wherein the 100 year window contains dates that span

in the 21st century (2000-2024) (p 248, right hand column). Ohms also discloses the step of

determining a century designator C.sub. I C.sub.2 for each date in the database, C.sub.I C.sub.2

having a first value if Y.sub.I Y.sub.2 is less than Y.sub.A Y.sub.B and having a second value if

Y.sub.I Y.sub.2 is equal to or greater than Y.sub.A Y.sub.B' by indicating that years that are later or

equal (25-99) to the pivot date (25) fall within the 20th century thereby equating C 1 C2 to 19 (L e.

1925-1999), whereas dates that are earlier 900-24) than the pivot date fall within the 21st century

thereby equating C1 C2 to 20 (i.e. 2000-2024) (see p 248, right-hand column. Additionally, OHMS

discloses the step of reformatting each date in the form C.sub.I C.sub.2 Y.sub. I Y.sub.2 M.sub. I

M.sub.2 D.sub. I D.sub.2 to facilitate further processing of the dates' by indicating that upon

determining that a two-digit date falls within the 20th or the 21st the century, it is expressed in

accordance with its corresponding century (i.e. 25-99 -----> 1925-1999, and 00-24 -----> 2000-2024)

(p 248, right hand column).

Regarding the step of selecting a 10-decade window with a Y.sub.A Y.sub.B value for the first decade

of the window, Y.sub.A Y.sub.B being no later than the earliest Y.sub. I Y.sub.2 year designator in the

database,' Ohms discloses specifying a year as the desired starting point (pivot date) of the 100 year

window (p 248, right hand column, 2nd paragraph).Ohms does not particularly detail that the pivot

date is earlier than the earliest two digit date in the database. Hazama, however, discloses an

analogous date processing system wherein, for a 100 year window, the pivot date for the window is

selected based on a two digit year date that is smaller than the smallest two digit year date in the

database (e.g. if smallest two digit date stored in database is 73, the pivot date for the 100 year

window is chosen to be 72). (page 4 of translated document, last paragraph). It would have been

obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited

references. The ordinary skilled artisan having read Ohms would immediately see the need to

determine which 100 year span to use. This determination would have led the ordinary skilled artisan

Application/Control Number ●9/512,592; 90/005,592; 90/005,62●●0/005727;   Page
90/006,541   115
Art Unit: 2161

to the Hazama reference, which teaches the pivot date being smaller than the smallest two digit date in the database having all the dates within a 100 year period as a solution to restrict the selection of Ohms's window and thereby forcing all dates already stored in the database to fall in the 20th century. Further, Ohms and Hazama do not specifically, disclose the step of ' sorting the dates in the form C. sub. I C.sub.2 Y. sub. I Y.sub.2 M. sub. I M.sub.2 D. sub. I D.sub.2.' However, Booth discloses an analogous system that utilizes the Clipper programming language to process dates stored in a database to thereby derive other dates therefrom (p 939, lines 1-3 et seq. In particular, analogously to Ohms and Hazama, Booth discloses the SET EPOCH command for comparing an entered two digit date with the year digit (pivot date) of the epoch setting to determine the century to place the date into (p 941, see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a corresponding eight digit date (C 1 C2YI Y2M 1 M21) 1132) (see p. 940-941). Additionally, Booth complements

Ohms and Hazama by suggesting the sorting of converted dates after having been reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary skill in the art of data processing at the time o f the instant invention to combine the teachings of the cited references. Booth's teaching of sorted reformatted dates would facilitate the Ohms-Hazama's system to return the reformatted dates in chronological sequence. And it would therefore be very useful when indexing the database in date order, as suggested by Booth in page 945.

As to claim 12, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of claim 11. Additionally, Ohms discloses step of' converting pre-existing date information having a different format into the format wherein M.sub. I M.sub.2 is the numerical month designator, D.sub. I

D.sub.2 is the numerical day designator and Y.sub.I Y.sub.2 is the numerical year designator' by

suggesting; the conversion from of a date from a Gregorian format (MMDDYYYY) to a short

Gregorian format (YYMMDD), wherein YY indicates the year, MM indicates the month and DD

indicates the date (p 247, table 1).

As to claim 13, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 11. Booth further complements Ohms and Hazama by disclosing the claimed 'the step of

selecting Y.sub.A Y.sub.B such that Y.sub.B is 0 (zero) ' by suggesting that the pivot date be set to 90

and by selecting set epoch to be 1990 (i.e. YAYB = 90), such that YB equals to zero (p. 942). It would

have been obvious to one of ordinary skill in the art of data processing at the time of the instant

invention to combine the teachings of the cited references. Booth's teaching of setting the pivot date

to a predetermined value would enable users of the Ohms-Hazama's system to have direct control

over the return of reformatted dates to thereby preset the date processing system in accordance with

their needs.

As to claim 14, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 11. Booth further complements Ohms and Hazama by disclosing the step of storing the sorted

dates and their associated information back into the database' by renaming and storing sorted dates

in the CUSTMER.DBF NEW CUST.DBF databases (p. 841). 1t would have been obvious to one of

ordinary skill in the art of data processing to combine the teachings of the cited references. Booth's

teaching of the storing the sorted reformatted dates would users of the Ohms-Hazama's system to

readily retrieve the reformatted dates in their chronological sequence at any time and in accordance

with their needs.

As to claim 15, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 14. Additionally, Booth discloses the step of 'manipulating information in the database having

the reformatted date therein' by renaming and storing the sorted dates in the CUSTMER.DBF NEW

CUST.DBF databases (p. 841). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. Booth's teaching of the storing the

sorted reformatted dates would users of the Ohms-Hazama's system to readily retrieve the

reformatted dates in their chronological sequence at any time and in accordance with their needs.

39. Claims 16-18, 20, 22, 24-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Ohms in view of Hazama. As to claim 16, Ohms substantially discloses the invention including the

claimed 'method of processing symbolic representations of dates stored in a database' by presenting

a computer-implemented method for processing date outside the twentieth century (see title, p 244 et

seq). In particular, Ohms discloses the step of 'providing a database with symbolic representations of

dates stored therein according to a format wherein M1, M2 is the numerical month designator, D1 D2

is the numerical day designator, and Y1 Y2 is the numerical year designator, all of the symbolic

representations of dates falling within a 10-decade period of time' by detailing a short Gregorian

format (MMDDYY) to represent dates (p 247, see table 1), wherein said dates fall within a 10 decade

or a 100 year window (p 249, left-hand column, lines 3-7 et seq). Ohms also discloses the step of

'determining a century designator C1 C2 for each symbolic representation of a date in the database.

CI C2 having a first value if YI Y2 is less than YAYB and having a second value if YIY2 is equal to or

greater than YAYB' by indicating that years that are later or equal (25-99) to the pivot date (25) would

fall within the 20th century thereby equating C1 C2 to 19 (Le. 1925-1999), whereas dates that are

earlier 900-24) than the pivot date would fall within the 21st century thereby equating C 1 C2 to 20 (i.

e. 2000-2024) (see p 248, right-hand column. Finally, Ohms discloses the step of 'reformatting the

symbolic representation of each symbolic representation of a date in the database, without

the addition of any new data field to the database, with the reformatted symbolic

representation of each date in the database having the values CI C2, YI Y2, MI M2, and DI

D2, in order to facilitate collectively further processing the reformatted symbolic

representations of each of the symbolic representations of each of the dates' by indicating that

upon determining that a two-digit date falls within the 20th or the 21st the century51, it is

expressed in accordance with its corresponding century without actually adding a new data

field or modifying the database (Le. 25-99 ----> 1925-1999, and 00-24 ----> 2000-2024) (p

248, right hand column). Regarding the step of 'selecting a window with a

YAYB value for a pivot date of the window, YAYB being no later than the earliest Y1 Y2

year designator in the database,' Ohms discloses specifying a year as the desired starting

point (pivot date) of the 100 year window (p 248, right hand column, 2nd paragraph). Ohms

does not particularly detail that the pivot date is earlier than the earliest two digit date in the

database. Hazama, however, discloses an analogous date processing system wherein, for

a 100 year window, the pivot date for the window is selected based on a two digit year date

that is smaller than the smallest two digit year date in the database (e.g. if smallest two digit

date stored in database is 73, the pivot date for the 100 year window is chosen to be 72).

(page 4 of translated document, last paragraph). It would have been obvious to one of

ordinary skill in the art of data processing to combine the teachings of the cited references.

5'01uns implicitly discloses that CIC2 corresponds to 19 or 20 depending on whether the date is less

than or greater than or equal to the pivot date.

The ordinary skilled artisan having read Ohms would immediately see the need to determine which 100 year span to use. This determination would have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date being smaller than the smallest two digit date in the database having all the dates within a 100 year period as a solution to restrict the selection of Ohms's window and thereby forcing all dates already stored in the database to fall in the 20th century.

As to claim 17, Ohms and Hazama disclose the invention as discussed in the rejection of claim 16. Additionally, Ohms discloses that 'the 10-decade window includes at least a portion of the decade beginning in the year 2000' by indicating that the 100 year window contains dates that span in the 21st century (2000-2024) (p 248, right hand column).

As to claim 18, Ohms and Hazama disclose the invention as discussed in the rejection of claim 17. Additionally, Ohms discloses that the step of determining includes the step of determining the first value as 20 and the second value as 19' by indicating that dates that are greater or equal to the pivot date fall within the 20th century (C1 C2=19) and dates that are less than the pivot date fall within the 21st century (C 1 C'.2=20) (p 248, right hand column).

As to claim 20, Ohms and Hazama disclose the invention as discussed in the rejection of claim 16. Additionally, Ohms discloses that the step of 'reformatting includes the step of reformatting each symbolic representation of a date into the format C1 C2 Y1 Y2 MI M2 DI D2 separately from the symbolic representations in the database' by indicating that dates that fall within the 20th century (greater than or equal to the pivot date) are preceded by 19 (e.g. 1925-1999), whereas dates that fall within the 21st century (less than the pivot date) are preceded by 20 (e.g. 2000-2024) (p 247, right hand column).

As to claim 22, Ohms and Hazama disclose the invention as discussed in the rejection of claim 16.

Additionally, Ohms discloses that the step of 'providing a database includes the step of converting

pre-existing date information having a different format into the format wherein M 1 M2 is the

numerical month designator, D 1 D2 is the numerical day designator and YI Y2 is the numerical year

designator' by suggesting the conversion from of a date from a Gregorian format (MIVIDDYYYY) to a

short Gregorian format (YYMMDD), wherein YY indicates the year, MM indicates the month and DD

indicates the date (p 247, table 1).

As to claim 24, Ohms and Hazama disclose the invention as discussed in the rejection of claim 16.

Additionally, Ohms discloses, 'after the step of reformatting, the storing the symbolic representation of

dates and their associated information back into the database' by suggesting that converted eight

dates are stored in the database although they take up more memory space than non-converted six

digits dates (p 249, left hand column).

As to claim 25, Ohms and Hazama disclose the invention as discussed in the rejection of claim 24.

Additionally, Ohms discloses, 'after the step of reformatting, the manipulating of information in the

database having the reformatted date information therein' by suggesting that the converted dates can

be saved in the database (p 249, left hand column).

the values C1 C2, YI Y2, in order to facilitate collectively further processing the reformatted symbolic

representations of each of the symbolic representations of each of the dates' by indicating that upon

determining that a two-digit date falls within the 20th or the 21st the century 52, it is expressed in

accordance with its corresponding century without actually adding a new data field or modifying the

database (i.e. 25-99 -----> 1925-1999, and 00-24 ----> 2000-2024) (p 248, right hand column).

Regarding the step of 'selecting A window with a YAYB value for the first decade of the window,

YAYB being no later than the earliest YI Y2 year designator in the database,' Ohms discloses

specifying a year as the desired starting; point (pivot date) of the 100 year window (p 248, right hand

column, 2n.d paragraph). Ohms does not particularly detail that the pivot date is earlier than the

earliest two digit date in the database. Hazama, however, discloses an analogous date processing

system wherein, for a 100 year window, the pivot date for the window is selected based on a two digit

year date that is smaller than the smallest two digit year date in the database (e.g. if smallest two digit

date stored in database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of

translated document, last paragraph). It would have been obvious to one of ordinary skill in the art of

data processing to combine the teachings of the cited references. The ordinary skilled artisan having

read Ohms would immediately see the need to determine which 100 year span to use. This

determination would have led the ordinary skilled artisan to the Hazama reference, which teaches the

pivot date being smaller than the smallest two digit date in the database having all

Ohms implicitly discloses that CIC2 corresponds to 19 or 20 depending on whether the date is less

than or greater than or equal to the pivot date.

40. Claims 19, 21, 23 are rejected under 35 U.S.C. 1 D3(a) as being unpatentable over Ohms in view

of Hazama, as applied to the rejection of claims 16-18, 20, 22, 24-25 above, further in view of Booth.

As to claim 19, Ohms and Hazama substantially disclose the invention as discussed in the rejection

of claim 16 above. Ohms and Hazama do not particularly disclose the additional step of `sorting the

symbolic representations of dates, after the step of reformatting.' However, Booth discloses an

analogous system that utilizes the Clipper programming language to process dates stored in a

database to thereby derive other dates therefrom (p 939, lines 1-3 et seq. In particular, analogously to

Ohms, Booth discloses the SET EPOCH command for comparing an entered two digit date with the

year digit (pivot date) of the epoch setting to determine the century to place the date into (p 941, see

SET EPOCH paragraph et seq) thereby converting a six digit date (M.MDDYY) into a corresponding

eight digit date (CIC2YIY2MIM21) I D2) (see p. 940-941). Additionally, Booth complements Ohms and

Hazama by suggesting the sorting of converted dates after having been reformatted by the SET

EPOCH command (p. 945). It would have been obvious to one of ordinary skill in the art of data

processing at the time of the instant invention to combine the teachings of the cited references.

Booth's teaching of sorted reformatted dates would facilitate the Ohms-Hazama's system to return the

reformatted dates in chronological sequence. And, it would therefore be very useful when indexing

the database in date order, as suggested by Booth in page 945.

As to claim 21, Ohms and Hazama substantially disclose the invention as discussed in the rejection

of claim 16 above. Ohms and Hazama do not specifically, disclose the additional

step of 'sorting the symbolic representations of dates using a numerical-order sort, after the step of

reformatting.' However, Booth discloses an analogous system that utilizes the Clipper programming

language to process dates stored in a database to thereby derive other dates therefrom (p 939, lines

1-3 et seq. In particular, analogously to Ohms, Booth discloses the SET EPOCH command for

comparing an entered two digit date with the year digit (pivot date) of the epoch setting to determine

the century to place the date into (p 941, see SET EPOCH paragraph et seq) thereby converting a six

digit date (MMDDYY) into A corresponding eight digit date (CIC2YIY2MIM2DID2) (see p. 940-941).

Additionally, Booth complements Ohms and Hazama by suggesting the sorting of converted dates

after having been reformatted by the SET EPOCH command (p. 945). It would have been obvious to

one of ordinary skill in the art of data processing at the time of the instant invention to combine the

teachings of the cited references. Booth's teaching of sorted reformatted dates would facilitate the

Ohms-Hazama's system to return the reformatted dates in chronological sequence. And, it would

therefore be very useful when indexing the database in date order, as suggested by Booth in page

945.

As to claim 23, Ohms and Hazama substantially disclose the invention as discussed in the rejection

of claim 16. Ohms and Hazama do not specifically, disclose the step of 'selecting includes the step of

selecting YAYB such that YB is 0 (zero).' However, Booth discloses an analogous system that utilizes

the Clipper programming language to process dates stored in a database to thereby derive other

dates therefrom (p 939, lines 1-3 et seq.) In particular, analogously to Ohms and Hazama, Booth

discloses the SET EPOCH command for comparing an entered two digit date with the year digit (pivot

date) of the epoch setting to determine the century to place the date into (p 941, see SET EPOCH

paragraph et seq) thereby converting a six digit date (MMDDYY) into a corresponding eight digit date

(CIC2YIY2MIM2DID2) (see p. 940-941). Additionally, Booth complements Ohms and

Hazama by suggesting that the pivot date be set to 90 by selecting set epoch to be 1990 (i.e.

YAYB = 90), such that YB equals to zero (p. 942). It would have been obvious to one of

ordinary skill in the art of data processing at the time of the instant invention to combine the

teachings of the cited references. Booth's teaching of setting the pivot date to a

predetermined value would enable users of the Ohms-Hazama's system to have direct

control over the return of reformatted dates to thereby preset the date processing system in

accordance with their needs.

41. Claims 26-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in

view of Hazama, further in view of Booth.

As to claim 26, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed 'method of processing dates in a database by presenting a computer implemented method

for processing date outside the twentieth century (see title, p 244 et seq). Ohms further discloses the

step of 'providing a database with dates stored therein according to a format wherein M 1 M2 is the

numerical month designator, D 1 D2 is the numerical day designator, and Y1 Y2 is the numerical year

designator, all of the symbolic representations of dates falling within a 10-decade period of time' by

detailing a short Gregorian format (MMDDYY) to represent dates (p 247, see table 1), wherein said

dates fall within a 10 decade or a 100 year window (p 249, left-sand column, lines 3-7 et seq) and

wherein the 100 yeas- window contains dates that span in the 21st century (2000-2024) (p 248, right

hand column). Ohms also discloses the step of 'determining a century designator C1 C2 for each date

in the database, CI C2 having a first value if YI Y2 is less than YAYB and having a second value if Y1

Y2 is equal to or greater than YAYB' by indicating that years that are later or equal (25-99) to the pivot

date (25) fall within the 20th century thereby equating C1 C2 to 19 (i.e. 1925-1999).. whereas dates

that are earlier 900-24) than the pivot date fall within the 21st century thereby equating C 1 C2 to 20

(i. e. 2000-2024) (see p 248, right-hand column. Additionally, Ohms discloses the step of 'reformatting

the symbolic representation of each symbolic representation of a date in the database, without the

addition of any new data field to the database, with the reformatted symbolic representation of each

date in the database having the values C 1 C2Y 1 Y2 M I M2, and D 1 D2, in order to facilitate

collectively further processing the reformatted symbolic representations of each of the symbolic

representations of each of the dates' by indicating that upon determining that a two digit date falls

within the 20th or the 21st the century, it is expressed in accordance with its corresponding century

without actually adding a new data field or modifying the database (i.e. 25-99 -----> 1925-1999, and

00-24 -----> 2000-2024) (p 248, right hand column).

Regarding the step of 'selecting a window with a YAYB value for a pivot date of the window, YAYB

being no later than the earliest Y1 Y2 year designator in the database,' Ohms discloses specifying a

year as the desired starting point (pivot date) of the 100 year window (p 248, right hand column, 2nd

paragraph). Ohms does not particularly detail that the pivot date is earlier than the earliest two digit

date in the database. Hazama, however, discloses an analogous date processing system wherein. for

a 100 year window, the pivot date for the window is selected based on a two digit year date that is

smaller than the smallest two digit year date in the database (e.g. if smallest two digit date stored in

database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated

document, last paragraph). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. The ordinary skilled artisan having read

Ohms would immediately see the need to determine which 100 year span to use. This determination

would have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date

being smaller than the smallest two digit date in the database having all the dates within a 100 year

period as a solution to restrict the selection of Ohms's window and thereby forcing all dates already

stored in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of 'sorting the dates in the form CI

C2YIY2MI M2 DI D2' However, Booth discloses an analogous system that utilizes the Clipper

programming language to process dates stored in a database to thereby derive other dates therefrom

(p 939, lines 1-3 et seq.) In particular, analogously to Ohms and Hazama, Booth discloses the SET

EPOCH command for comparing an entered two digit date with the year digit (pivot date) of the epoch

Application/Control Number●9/512,592; 90/005,592; 90/005,62●●0/005727;   Page
90/006,541   126
Art Unit: 2161

setting to determine the century to place the date into (p 941, see SET EPOCH paragraph et seq)

thereby converting a six digit date (MMDDYY) into a corresponding eight digit date (C 1 C2Y 1 Y2M 1

M2D 1 D2) (see p. 940941). Additionally, Booth complements Ohms and Hazama by suggesting the

sorting of converted dates after having been reformatted by the SET EPOCH command (p. 945). It

would have been obvious to one of ordinary skill in the art of data processing at the time of

the instant invention to combine the teachings of the cited references. Booth's teaching of

sorted reformatted dates would facilitate the Ohms-Hazama's system to return the

reformatted dates in chronological sequence. And it would therefore be very useful when

indexing the database in date order, as suggested by Booth in page 945.

As to claim 27, Ohms, Hazama and Booth disclose the invention as discussed in the

rejection of claim 26. Additionally, Ohms discloses step of converting pre-existing date

information having a different format into the format wherein MI M2 is the numerical month

designator, DI D2 is the numerical day designator and Y1 Y2 is the numerical year designator' by

suggesting the conversion from of a date from a Gregorian format (MMDDYYYY) to a short Gregorian

format (YYMMDD), wherein YY indicates the year, MM indicates the month and DD indicates the date

(p ::47, table 1).

As to claim 28, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 26. Booth further complements Ohms and Hazama by disclosing the claimed the step of

'selecting includes the step of selecting YAYB such that YB is 0 (zero)' by suggesting that the pivot

date be set to 90 and by selecting set epoch to be 1990 (i.e. YAYB = 90), such that YB equals to zero

(p. 942). It would have been obvious to one of ordinary skill in the art of data processing at the time of

the instant invention to combine the teachings of the cited references. Booth's teaching of setting the

pivot date to a predetermined value would enable users of the Ohms-Hazama's system to have direct

control over the return of reformatted dates to thereby preset the date processing system in

accordance with their needs.

As to claim 29, Ohms, Hazama and Booth disclose the invention as discussed in the

rejection of claim 26. Booth further complements Ohms and Hazama by disclosing, after

the step of sorting, the step of 'storing the sorted dates and their associated information back

into the database' by renaming and storing sorted dates in the CUSTMER.DBF

NEW CUST.DBF databases (p. 841). It would have been obvious to one of ordinary skill in the art of

data processing to combine the teachings of the cited references. Booth's teaching of the storing the

sorted reformatted dates would users of the Ohms-Hazama's system to readily retrieve the

reformatted dates in their chronological sequence at any time and in accordance with their needs.

As to claim 30, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 29. Additionally, Booth discloses the step of manipulating information in the database having

the reformatted dates therein' by renaming and storing the sorted dates in the CUSTMER.DBF NEW

CUST.DBF databases (1). 841). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. Booth's teaching of the storing the

sorted reformatted dates would users of the Ohms-Hazama's system to readily retrieve the

reformatted dates in their chronological sequence at any time and in accordance with their needs.

the values C1 C2, Y1 Y2, in order to facilitate collectively further processing the reformatted symbolic

representations of each of the symbolic representations of each of the dates' by indicating that upon

determining that a two-digit date fills within the 20th or the 21st the century52, it is expressed in

accordance with its corresponding century without actually adding a new data field or modifying the

database (i.e. 25-99 ----> 1925-1999, and 00-24 ----> 2000-2024) (p 248, right hand column).

Regarding the step of 'selecting A window with a YAYB value for the first decade of the window,

YAYB being no later than the earliest Y1 Y2 year designator in the database,' Ohms discloses

specifying a year as the desired starting point (pivot date) of the 100 year window (p 248, right hand

column, 2nd paragraph). Ohms does not particularly detail that the pivot date is earlier than the

earliest two digit date in the database. Hazama, however, discloses an analogous date processing

system wherein, for a 100 year window, the pivot date for the window is selected based on a two digit

year date that is smaller than the smallest two digit year date in the database (e.g. if smallest two digit

date stored in database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of

translated document, last paragraph). It would have been obvious to one of ordinary skill in the art of

data processing to combine the teachings of the cited references. The ordinary skilled artisan having

read Ohms would immediately see the need to determine which 100 year span to use. This

determination would have led the ordinary skilled artisan to the Hazama reference, which teaches the

pivot date being smaller than the smallest two digit date in the database having all the dates within a

100 year period as ;a solution to restrict the selection of Ohms's window and thereby forcing all dates

already stored in the database to fall in the 20th century.

Ohms implicitly discloses that CIC2 corresponds to 19 or 20 depending on whether the date is less

than or greater than or equal to the pivot date.

43. Claim 32 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 32, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed 'method of processing dates in a database' by presenting a computer implemented method

for processing date outside the twentieth century (see title, p 244 et seq). Ohms further discloses the

step of 'providing a database with symbolic representations of dates stored therein according to a

format wherein Y1 Y2 is the numerical year designator' by detailing a short Gregorian format

(MMDDYY) to represent dates (p 247, see table 1), wherein said dates fall within a 10 decade or A

100 year window (p 249, left-hand column, lines 3-7 et seq) and wherein the 100 ;year window

contains dates that span in the 21st century (2000-2024) (p 248, right hand column). Ohms also

discloses the step of 'determining a century designator C1 C2 for each symbolic representation of a

date in the database, C1 C2 having a first value if Y1 Y2 is less than YAYB and having a second

value if Y1 Y2, is equal to or greater than YAYB' by indicating that years that are later or equal

(25-99) to the pivot date (25) fall within the 20th century thereby equating C I C2 to 19 (i.e.

1925-1999), whereas dates that are earlier 900-24) than the pivot date fall within the 21st century

thereby equating C1 C2 to 20 (Le. 2000-2024) (see p 248, right-hand column. Additionally, Ohms

discloses the step of reformatting the symbolic representation of each of the dates in the database,

without the addition of any new data field to the database, with the reformatted symbolic

representation of each date in the database having the values C 1 C2, Y1 Y2, in order to facilitate

collectively further processing the reformatted symbolic representations of each of the dates' by

indicating that upon determining that a two-digit date falls within the 20th or the 21st the century, it is

expressed in accordance with its corresponding century without actually adding a new data field or

modifying the database (i.e. 25-99 -----> 1925-1999, and 00-24 -----> 2000-2024) (p 248, right hand

column).

Regarding the step of `selecting a window with a YAYB value for a pivot year of the window, YAYB

being no later than the earliest Y1 Y2 year designator in the database,' Ohms discloses specifying a

year as the desired starting point (pivot date) of the 100 year window (p 248, right hand column, 2nd

paragraph). Ohms does not particularly detail that the pivot date is earlier than the earliest two digit

date in the database. Hazama, however, discloses an analogous date processing system wherein, for

a 100 year window, the pivot date for the window is selected based on a two digit year date that is

smaller than the smallest two digit year date in the database (e g. if smallest two digit date stored in

database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated

document, last paragraph). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. Hazama's teaching of the pivot date

being smaller than the smallest two digit date in the database would restrict the selection of Ohms's

window and thereby forcing all dates already stored in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of sorting the dates in the form C1

C2 Y1 Y2' However, Booth discloses an analogous system that utilizes

the Clipper programming language to process dates stored in a database to thereby derive other

dates therefrom (p 939, lines 1-3 et seq. In particular, analogously to Ohms and Hazama, Booth

discloses the SET EPOCH command for comparing an entered two digit date with the year digit (pivot

date) of the epoch setting to determine the century to place the date into (p 941, see SET EPOCH

paragraph et seq) thereby converting a six digit date (MMDDYY) into a corresponding eight digit date

(C 1 ~C2Y 1 Y2M 1 M21) 1 D2) (see p. 940941). Additionally, Booth complements Ohms and

Hazama by suggesting the sorting of converted dates after having been reformatted by the SET

EPOCH command (p. 945). It would have been obvious to one of ordinary skill in the art of data

processing at the time of the instant invention to combine the teachings of the cited references.

Booth's teaching of sorted reformatted dates would facilitate the Ohms-Ha2ama's system to return

the reformatted dates in chronological sequence. And it would therefore be very useful when indexing

the database in date order, as suggested by Booth in page 945.

44. Claim 33 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama.

As to claim 33, Ohms substantially discloses the invention including the claimed `method of

processing symbolic representations of dates stored in a database' by presenting A

computer-implemented method for processing date outside the twentieth century (see title, p 244 et

seq). In particular, Ohms discloses the step of 'providing A database with symbolic

representations of dates stored therein according to a format wherein Y 1 Y2 is the numerical year

designator' by detailing a short Gregorian format (.VIMDDYY) to represent dates (p 247, see table 1),

wherein said dates fall within a 10 decade or a 100 year window (p 249, left hand column, lines 3-7 et

seq). Ohms also discloses the step of 'determining a century designator CI C2 for each symbolic

representation of a date in the database, CI C2 having a first value if YI Y2 is less than YAYB and

having a second value if YI Y2 is equal to or greater than YAYB' by indicating that years that are lat--r

or equal (25-99) to the pivot date (25) would fall within the 20th century thereby equating C1 C2 to 19

(i. e. 1925-1999), whereas dates that are earlier 900-24) than the pivot date would fall within the 21st

century thereby equating C1 C2 to 20 (i.e. 2000-2024) (see p 248, right-hand column). Finally, Ohms

discloses the step of 'reformatting the symbolic representation of each symbolic representation of a

date in the database without changing any of the symbolic representations of a date in the database

during the reformatting step, with the reformatted symbolic representation of each date in the

database having the values C1C2Y1Y2, in order to facilitate collectively further processing the

reformatted symbolic representations of each of the dates' by indicating that upon determining that a

two-digit date falls within the 20th or the 21st the century53, it is expressed in accordance with its

corresponding century without actually adding a new data field or modifying the database (i.e. 25-99

----> 1925-1999, and 00-24 ----> 2000-2024) (p 248, right hand column).

"Ohms implicitly discloses that C1C2 corresponds to 19 or 20 depending on whether the date is less

than or greater than or equal to the pivot date.

Regarding the step of `selecting a window with a YAYB value for the first decade of the window,

YAYB being no later than the earliest Y1 Y2 year designator in the database,' Ohms discloses

specifying a year as the desired starting point (pivot date) of the 100 year window (p 248, right hand

column, 2nd paragraph). Ohms does not particularly detail that the pivot date is earlier than the

earliest two digit date in the database. Hazama, however, discloses an analogous date processing

system wherein, for a 100 year window, the pivot date for the window is selected based on a two digit

date that is smaller than the smallest two digit year date in the database (e.g. if smallest two digit date

stored in database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of

translated document, last paragraph). It would have been obvious to one of ordinary skill in the art of

data processing to combine the teachings of the cited references. The ordinary skilled artisan having

read Ohms would immediately see the need to determine which 100 year span to use. This

determination would have led the ordinary skilled artisan to the Hazama reference, which teaches the

pivot date being smaller than the smallest two digit date in the database having all the dates within a

100 year period as a solution to restrict the selection of Ohms's window and thereby forcing all dates

already stored in the database to fall in the 20th century.

45. Claims 34-59 are rejected under.i5 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 34, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed `method for representing and utilizing dates stored in at least one date field of a database

utilizing symbolic representations of the dates stored in the at least one date field of the database,

which are in a format that creates ambiguity between dates in each of a pair of adjacent centuries' by

presenting a computer-implemented method for processing date outside the twentieth century (see

title, p 244 et seq). Ohms further discloses the steps of `converting each of the symbolic

representations of dates stored in the at least one date field of the database to a symbolic

representation of each of the respective dates that does not create the ambiguity, by windowing the

symbolic representations of each of the respective dates as stored in the at least one date field of the

database against a pivot year represented by one of the symbolic representations of the dates as

stored in the at least one date field of the database, without the addition of any new data field to the

database for purposes of such windowing and converting; and running a program collectively on each

of the converted symbolic representations of each of the respective dates to manipulate the dates

represented by the converted symbolic representations, separately from the date data symbolic

representations contained in the at least one date field of the database' by detailing a short Gregorian

format (MMDDYY) to represent dates (p 2z.7, see table 1), wherein said dates fall within a 10 decade

or a 100 year window (p 249, left-hand column, lines 3-7 et seq) and wherein the 100 year window

contains dates that span :in the 21st century (2000-2024) (p 248, right hand column). Ohms also

indicates that years that are later or equal (25-99) to the pivot date (25) fall within the 20th century

thereby equating CIC2 to 19 (i.e. 1925-1999), whereas dates that are earlier 900-24) than the pivot

date fall within the 21st century thereby equating CIC2 to 20 (i.e. 2000-2024) (see p 248, right-hand

column. Additionally, Ohms discloses that upon determining that a two-digit date falls within the 20th

or the 21st the century, it is expressed in accordance with its corresponding century without actually

adding A new data field or modifying the database (i.e. 25-99 -----> 1925-1999, and 00-24 ----->

2000-2024) (p 248, right hand column). Regarding the selection of a pivot year for the century

window, Ohms discloses specifying a year as the desired starting point (pivot date) of the 100 year

window (p 248, right hand column, 2nd paragraph). Ohms does not particularly detail that the pivot

date is earlier than the earliest two digit date in the database. Hazama, however, discloses an

analogous date processing system wherein, for a 100 year window, the pivot date for the

window is selected based on a two digit year date that is smaller than the smallest two digit

year date in the database (e.g. if smallest two digit date stored in database is 73, the pivot

date for the 100 year window is chosen to be 72). (page 4 of translated document, last

paragraph). It would have been obvious to one of ordinary skill in the art of data processing

to combine the teachings of the cited references. The ordinary skilled artisan having read

Ohms would immediately see the need to determine which 100 year span to use. This

determination would have led the ordinary skilled artisan to the Hazama reference, which

teaches the pivot date being smaller than the smallest two digit date in the database having all the

dates within a 100 year period as a solution to restrict the selection of Ohms's window and thereby

forcing all dates already stored in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifica:1ly, disclose the step of sorting the dates in the form C1

C2 Y1 Y2' However, Booth discloses an analogous system that utilizes the Clipper programming

language to process dates stored in a database to thereby derive other dates therefrom (p 939, lines

1-3 et seq. In particular, analogously to Ohms and Hazama, Booth discloses the SET EPOCH

command for comparing an entered two digit date with the year digit (pivot date) of the Epoch setting

to determine the century to place the date into (p 941, see SET EPOCH paragraph et seq) thereby

converting a six digit date (MMDDYY) into a corresponding eight digit date (C1C2Y1Y2MIM2D1D2)

(seep. 940941). Additionally, Booth complements Ohms and Hazama by suggesting the sorting of

converted dates after having been reformatted by the S1JT EPOCH command (p. 945). It would have

been obvious to one of ordinary skill in the art of data processing at the time of the instant invention to

combine the teachings of the cited references. Booth's teaching of sorted reformatted dates would

facilitate the Ohms -Hazama's system to return the reformatted dates in chronological sequence. And

it would therefore be very useful when indexing the database in date order, a:; suggested by Booth in

page 945.

As to claim 35, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 34. Additionally, Ohms discloses step of 'opening the database prior to the step of converting'

by providing a subroutine to retrieve a six digit date from its storage location in an existing application

program (i.e. requires opening the DB, first) prior to converting said date to an eight digit format (p

248, right hand column et seq).

As to claim 36, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 34. Booth further complements Ohms and Hazama by disclosing the step of `collectively sorting

the converted symbolic representations prior to the step of running the program on the converted

symbolic representations' by suggesting that a string representation

can be used to sort and index the converted dates such that the dates appear in chronological order

(page 945 et seq).

As to claim 37, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 35. Booth further complements Ohms and Hazama by disclosing the step of 'collectively sorting

the converted symbolic representations prior to the step of running the program on the converted

symbolic representations' by suggesting that a string representation can be used to sort and index the

converted dates such that the dates appear in chronological order (page 945 et seq).

As to claim 38, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 34. Booth further complements Ohms and Hazama by disclosing the step of 'collectively

manipulating the converted symbolic representations prior to the step of running the program on the

converted symbolic representations' by suggesting a string representation can be used to sort and

index the converted dates such that the dates appear in chronological order (page 945 et seq).

As to claim 39, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 35. Booth further complements Ohms and Hazama by disclosing the step of 'collectively

manipulating the converted symbolic representations prior to the step of running the program on the

converted symbolic representations' by suggesting that a string representation can be used to sort

and index the converted dates such that the dates appear in chronological order (page 945 et seq).

As to claim 40, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 34. Booth further complements Ohn1; and Hazama by disclosing the step of `collectively sorting

the converted symbolic representations according to a different data field contained in the database

from the at least one date field, prior to the step of running the program on the converted symbolic

representations' by .suggesting that a string representation can be used to sort and index the

converted dates such that the dates appear in chronological order (page 945 et seq), whereby said

dates, contained in a field of the database, are sorted accordingly in a different field of the database

(page 839-40 et seq).

As to claim 41, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 35. Booth further complements Ohms and Hazama by disclosing the step of `collectively sorting

the converted symbolic representations according to a different data field contained in the database

from the at least one date field, prior to the step of running the program on the converted symbolic

representations' by suggesting that a string representation can be used to sort and index the

converted dates such that the dates appear in chronological order (page 945 et seq), whereby said

dates, contained in a field of the database, are sorted accordingly in a different field of the database

(page 839-40 et seq).

As to claim 42, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 34. Booth further complements Ohms and Hazama by disclosing the step of 'collectively

manipulating the converted symbolic representations according to A different data field contained in

the database from the at least one date field, prior to the step of running the program on the

converted symbolic representations' by suggesting that A string representation can be used to sort

and index the converted dates such that the dates appear in chronological order (page 945 et seq),

whereby said dates, contained in A field of the database, are sorted accordingly in A different field of

the database (page 839-40 et seq).

As to claim 43, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 35. Booth further complements Ohms and Hazama by disclosing the step of 'collectively

manipulating the converted symbolic representations according to a different data entry field

contained in the database from the at least one date field, prior to the step of running the program on

the converted symbolic representations' by suggesting that a string representation can be used to sort

and index the converted dates such that the dates appear in chronological order (page 945 et seq),

whereby said dates, contained in a field of the database, are sorted accordingly in a different field of

the database (page 839-40 et seq).

As to claim 44, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 34. Booth further complements Ohms and Hazama by disclosing the step of 'performing an

operation which manipulates the data in a data field associated with the at least one date field of the

database according to the converted symbolic representation of the date' by suggesting that a string

representation can be used to sort and index the converted dates such that the dates appear in

chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are

sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 45, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 35. Booth further complements Ohms and Hazama by disclosing the step of 'performing an

operation which manipulates the data in a data field associated with the at least one date field of the database according to the converted symbolic representation of the date' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order 'page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 46, Ohms, Hazama, and Booth disclose the invention as discussed in the rejection of claim 34. Ohms further discloses the step of 'converting at least a substantial portion of each of the plurality of symbolic representations of dates in the at least one date field and repeating this step until each of the date data entries in the at least one date field is converted into the format that does not have the ambiguity' by converting the current date stored in the database field from an ambiguous six digit format (YYMMDD) into an unambiguous 8-digit format (CCYYMMDD), wherein the century for the date is specified (p 248, right hand column et seq).

As to claim 47, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of claim 35. Ohms further discloses the step of 'converting at least a substantial portion of each of the plurality of symbolic representations of dates in the at least one date field and repeating this step until each of the date data entries in the at least one date field is converted into the format that does not have the ambiguity' by converting the current date stored in the database field from an ambiguous six digit format (YYMMDD) into an unambiguous 8-digit format (CCYYNIMDD), wherein the century for the date is specified (p 248, right hand column et seq).

As to claim 48, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 46. Booth further complements Ohms and Hazama by disclosing the step of 'collectively sorting

the converted symbolic representations prior to the step of running the program on the converted

symbolic representations' by suggesting that a string representation can be used to sort and index the

converted dates such that the dates appear in chronological order (page 945 et seq), whereby said

dates, contained in a field of the database, are sorted accordingly in a different field of the database

(page 8_'39-40 et seq).

As to claim 49, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 47. Booth further complements Ohms and Hazama by disclosing the step of 'collectively sorting

the converted symbolic representations prior to the step of running the program on the converted

symbolic representations' by suggesting that A string representation can be used to sort and index

the converted dates such that the dates appear in chronological order (page 945 et seq), whereby

said dates, contained in a field of the database, are sorted accordingly in a different field of the

database (page 839-40 et seq).

As to claim 50, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 46. Booth further complements Ohms and Hazama by disclosing the step of 'collectively

manipulating the converted symbolic representations' by suggesting that a string representation can

be used to sort and index the converted dates such that the dates appear in chronological order

(page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in

a different field of the database (page 839-40 et seq).

As to claim 51, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 49. Booth further complements Ohms and Hazama by disclosing the step of collectively

manipulating the converted symbolic representations' by suggesting that a string representation can

be used to sort and index the converted dates such that the dates appear in chronological order

(page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in

a different field of the database (page 839-40 et seq).

As to claim 52, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 46. Booth further complements Ohms and Hazama by disclosing the step of 'collectively sorting

the converted symbolic representations according to a different data field in the database than the at

least one date field, prior to the step of running the program by suggesting that a string representation

can be used to sort and index the converted dates such that the dates appear in chronological order

(page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in

a different field of the database (page 839-40 et seq).

As to claim 53, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 47. Booth further complements Ohms and Hazama by disclosing the step of `collectively sorting

the converted symbolic representations according to a different data field in the database than the at

least one date field, prior to the step of running the program' by suggesting that a string

representation can be used to sort and index the converted dates such that the dates appear in

chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are

sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 54, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 52. Booth further complements Ohms and Hazama by disclosing the step of `collectively

manipulating the converted symbolic representations' by suggesting that a string representation can

be used to sort and index the converted dates such that the dates appear in chronological order

(page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in

a different field of the database (page 839-40 et seq).

As to claim 55, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of

claim 53. Booth further complements Ohms and Hazama by disclosing the step of collectively

manipulating the converted symbolic representations' by suggesting that a string representation can

be used to sort and index the converted dates such that the dates appear in chronological order

(page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in

a different field of the database (page 839-40 et seq).

As to claim 56, Ohms, Hazama and Booth disclose the invention as discussed in the

rejection of claim 52. Booth further complements Ohms and Hazama by disclosing the step

of 'performing an operation which manipulates the data in a data field associated with the at

least one date field of the database according to the converted symbolic representation of the date' by

suggesting that a string representation can be used to sort and index the converted dates such that

the dates appear in chronological order (page 945 et seq), whereby said dates, contained in A field of

the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 57, Ohms, Hazama, and Booth disclose the invention as discussed in the rejection of claim 53. Booth further complements Ohms and Hazama by disclosing the step of 'performing an operation which manipulates the data in a data field associated with the at least one date field of the database according to the converted symbolic representation of the date' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).Application/Control

As to claim 58, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of claim 54. Booth further complements Ohms and Hazama by disclosing the step of 'performing an operation which manipulates the data in a data field associated with the at least one date field of the database according to the converted symbolic representation of the date' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

As to claim 59, Ohms, Hazama and Booth disclose the invention as discussed in the rejection of claim 55. Booth further complements Ohms and Hazama by disclosing the step of 'performing an operation which manipulates the data. in a data field associated with the at least one date field of the database according to the converted symbolic representation of the date' by suggesting that a string representation can be used to sort and index the converted dates such that the dates appear in chronological order (page 945 et seq), whereby said dates, contained in a field of the database, are sorted accordingly in a different field of the database (page 839-40 et seq).

46. Claim 60 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 60, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed `method for representing and utilizing dates stored in at least one date field of A database

utilizing symbolic representations of the dates stored in the at least one date field of the database,

which are in a format that creates ambiguity between dates in each of a pair of adjacent centuries' by

presenting a computer-implemented method for processing date outside the twentieth century (see

title, p 244 et seq). Ohms further discloses the steps of `converting each of the symbolic

representations of dates stored in the at least one date field of the database to symbolic

representation of each of the respective dates that does not create the ambiguity, by windowing the

symbolic representations of each of the respective dates as stored in the at least one date field of the

database against a pivot year represented by one of the symbolic representations of the dates as

stored in the at least one date field of the database, without modifying any of the symbolic

representations of dates in the at least one date field of the database for purposes of such windowing

and converting; running a program on each of the converted symbolic representations of each of the

respective dates to manipulate data in the database according to the dates represented by the

converted symbolic representations, separately from the date data symbolic representations of dates

contained in the at least one date field of the database' by detailing a short Gregorian format

(MMDDYY) to represent dates (p 247, see table 1), wherein said dates fall within a 10 decade or a

100 year window (p 249, left-hand column, lines 3-7 et seq) and wherein the 100 year window

contains dates that span in the 21st century (2000-2024) (p 248, right hand column). Ohms also

indicates that years that are later or equal (25-99) to the pivot date (25) fall within the 20th century

thereby equating C1 C2 to 19 (i.e. 1925-19)9), whereas dates that are earlier 900-24) than the pivot

date fall within the 21st century thereby equating CIC2 to 20 (i.e. 2000-2024) (see p 248, right-hand

column. Additionally, Ohms discloses that upon determining that a two-digit date falls within the 20th

or the 21st the century, it is expressed in accordance with its corresponding century without actually

adding a new data field or modifying the database (i.e. 25-99 ----> 1925-1999, and 00-24 ----->

2000-2024) (p 248, right hand column).

Regarding the selection of A. pivot year for the century window, Ohms discloses specifying a year as

the desired starting point (pivot date) of the 100 year window (p 248, right hand column, 2nd

paragraph). Ohms does not particularly detail that the pivot date is earlier than the earliest two digit

date in the database. Hazama, however, discloses an analogous date processing system wherein, for

a 100 year window, the pivot date for the window is selected based on a two digit year date that is

smaller than the smallest two digit year date in the database (e.g. if smallest two digit date, stored in

database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated

document, last paragraph). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. The ordinary skilled artisan having read

Ohms would immediately see the need to determine which 100 year span to use. This determination

would have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date

being smaller than the smallest two digit date in the database having all the dates within a 100 year

period as a solution to restrict the selection of Ohms's window and thereby forcing all dates already

stored in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of sorting the dates in the form C1

C2 YI Y2' However, Booth discloses an analogous system that utilizes the Clipper programming

language to process dates stored in a database to thereby derive other dates therefrom (p 939, lines

1-3 et seq. In particular, analogously to Ohms and Hazama, Booth discloses the SET EPOCH

command far comparing an entered two digit date with the year digit (pivot date) of the epoch setting

to determine the century to place the date into (p 941, see SET EPOCH paragraph et seq) thereby

converting a six digit date (MM DDYY) into a corresponding eight digit date (C 1 C2Y 1 Y2 M 1 M2D 1

D2) (see p. 940941). Additionally, Booth complements Ohms and Hazama by suggesting the sorting

of converted dates after having been reformatted by the SET EPOCH command (p. 945). It would

have been obvious to one of ordinary skill in the art of data processing at the time of the instant

invention to combine the teachings of the cited references. Booth's teaching of sorted reformatted

dates would facilitate the Ohms-Hazama's system to return the reformatted dates in chronological

sequence. And it would therefore be very useful when indexing the database in date order, as

suggested by Booth in page 945.

47. Claim 61 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 61, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed 'method for representing and utilizing dates stored in at least one date field of a database

utilizing symbolic representations of the dates stored in the at least one date field of the database,

which are in a format that creates ambiguity between dates in each pair of adjacent centuries' by

presenting a computer- implemented method for processing date outside the twentieth century (see

title, p 244 et seq). Ohms further discloses the steps of `converting each of the symbolic

representations of dates stored in the at least one date field of the database to a symbolic

representation of each of the respective dates that does not create the ambiguity, by windowing the

symbolic representations of each of the respective dates as stored in the at least one date field of the

database against a pivot year represented by one of the symbolic representations of the dates as

stored in the at least one date field of the database, without modifying any of the symbolic

representations of dates in the at least one date field of the database for purposes of such windowing

and converting; and running a program collectively on each of the converted symbolic representations

of each of the respective dates to manipulate the dates represented by the converted symbolic

representations, separately from the symbolic representations of dates contained in the at least one

date field of the database' by detailing a short Gregorian format (MMDDYY) to represent dates (p

247, see table 1), wherein said dates fall within a 10 decade or A 100 year window (p 249, left-hand

column, lines 3-7 et seq) and wherein the 100 year window contains dates that span in the 21st

century (2000-2024) (p 248, right hand column). Ohms also indicates that years that are later or equal

(25-99) to the pivot date (25) fall within the 20th century thereby equating C1 C2 to 19 (i.e.

1925-1999), whereas dates that are earlier 900-24) than the pivot date fall within the 21st century

thereby equating C 1 C2 to 20 (i. e. 2000-2024) (see p 248, right-hand column. Additionally, Ohms

discloses that upon determining that a two-digit date falls within the 20th or the 21st the century, it is

expressed in accordance with its corresponding century without actually adding a new data field or

modifying the database (i.e. 25-99 -----> 1925-1999, and 00-24 -----> 2000-2024) (p 248,

right hand column).

Regarding the selection of a pivot year for the century window, Ohms discloses

of the 100 year window (p 248, specifying a year as the desired starting point (pivot date I

right hand column, 2nd paragraph). Ohms does not particularly detail that the pivot date is earlier than

the earliest two digit date in the database. Hazama, however, discloses an analogous date

processing system wherein, for a 100 year window, the pivot date for the window is selected based

on a two digit year date that is smaller than the smallest two digit year date in the database (e.g. if

smallest two digit date: stored in database is 73, the pivot date for the 100 year window is chosen to

be 72). (page 4 of translated document, last paragraph). It would have been obvious to one of

ordinary skill in the art of data processing to combine the teachings of the cited references. The

ordinary skilled artisan having read Ohms would immediately see the need to determine which 100

year span to use. This determination would have led the ordinary skilled artisan to the Hazama

reference, which teaches the pivot date being smaller than the smallest two digit date in the database

having all the dates within a 100 year period as a solution to restrict the selection of Ohms's window

and thereby forcing all dates already stored in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of sorting the dates in the form C1

C2 YJ Y2' However, Booth discloses an analogous system that utilizes the Clipper programming

language to process dates stored in a database to thereby derive other dates therefrom (p 939, lines

1-3 et seq. In particular, analogously to Ohms and Hazama, Booth discloses the SET EPOCH

command for comparing an entered two digit date with the year digit (pivot date) of the epoch setting

to determine the century to place the date into (p 941, see SET EPOCH paragraph et seq) thereby

converting a six digit date (MMDDYY) into a corresponding eight digit date (C 1 C2Y 1 Y2M 1 M2D 1

D2) (see p. 940941). Additionally, Booth complements Ohms and Ha2ama by suggesting the sorting

of converted dates after having been reformatted by the SST EPOCH command (p. 945). It would

have been obvious to one of ordinary skill in the art of data processing at the time of the instant

invention to combine the teachings of the cited references. Booth's teaching of sorted reformatted

dates would facilitate the Ohms-Hazama's system to return the reformatted dates in chronological

sequence. And it would therefore be very useful when indexing the database in date order, as

suggested by Booth in page 945.

48. Claim 62 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 62, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed 'method for representing and utilizing dates stored in at least one date field of a database

utilizing symbolic :representations o f the dates stored in the at least one date field of the database,

which are in a format that creates ambiguity between dates in each of a pair of adjacent centuries' by

presenting a computer-implemented method for processing date outside the twentieth century (see

title, p 244 et seq). Ohms further discloses the steps of 'converting each of the symbolic

representations of dates stored in the at least one date field of the database to a symbolic

representation of each of the respective dates that does not create the ambiguity, by windowing the

symbolic representations of each of the respective dates as stored in the at least one date field of the

database against a pivot year represented by one of the symbolic representations or the dates as

stored in the at least one date field of the database, without the addition of any new data field to the

database for purposes of such windowing and converting; and running a program on -:he stored

converted symbolic representations to manipulate data in the database according to the dates

represented by the converted symbolic representations, separately from the° symbolic

representations of dates contained in the at least one date field of the database' ~by detailing a short

Gregorian format (MMDDYY) to represent dates (p 247, see table 1), wherein said dates fall within a

10 decade or A 100 year window (p 249, left-hand column, lines 3-7 et seq) and wherein the 100 year

window contains dates that span in the 21st century (2000-2024) (p 248, right hand column). Ohms

also indicates that years that are later or equal (25-99) to the pivot date (25) fall within the 20th

century thereby equating C 1 C2 to :19 (i. e. 1925-1999), whereas dates that are earlier 900-24) than

the pivot date fall within the 21st century thereby equating C 1 C2 to 20 (i.e. 2000-2024) (see p 248,

right-hand column. Additionally, Ohms discloses that upon determining that a two-digit date falls

within the 20th or the 21st the century, it is expressed in accordance with its corresponding century

without actually adding a new data field or modifying the database (i.e. 25-99 -----> 1925-1999, and

00-24 -----> 2000-2024) (p 248, right hand column). Regarding the selection of a pivot year for the

century window, Ohms discloses 'specifying a year as the desired starting point (pivot date) of the

100 year window (p 248, right hand column, 2nd paragraph). Ohms does not particularly detail that

the pivot date is earlier than the earliest two digit date in the database. Hazama, however, discloses

an analogous date processing system wherein, for a 100 year window, the pivot date for the window

is selected based on a two digit year date that is smaller than the smallest two digit year date in the

database (e.g. if smallest two digit date stored in database is 73, the pivot date for the 100 year

window is chosen to be 72). (page 4 of translated document, last paragraph). It would have been

obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited

:references. Hazama's teaching of the pivot date being smaller than the smallest two digit date in the

database would restrict the selection of Ohms's window and thereby forcing all dates already stored

in the database to fall in the 20th century. Further, Ohms and Hazama do not specifically, disclose

the step of sorting the dates in the form C1 C2 Y1 Y2' However, Booth discloses an analogous

system that utilizes the Clipper programming language to process dates stored in a database to

thereby derive other dates therefrom (p 939, lines 1-3 et seq. In particular, analogously to Ohms and

Hazama, Booth discloses the SET EPOCH command for comparing an entered two digit date with

the year digit (pivot date) of the epoch setting to determine the century to place the date into (p 941,

see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a

corresponding eight digit date (C] C2YIY2MIM2DID2) (see p. 940941). Additionally, Booth

complements Ohms and Hazama by suggesting the sorting of converted dates after having been

reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary

skill in the art of data processing at the time of the instant invention to combine the teachings of the

cited references. Booth's teaching of sorted reformatted dates would facilitate the Ohms-Hazama's

system to return the reformatted dates in chronological sequence. And it would therefore be very

useful when indexing the database in date order, as suggested by Booth in page 945.

Ohms and Hazama do not specifically disclose the step of 'storing the converted

symbolic representations separate from the at least one date field of the database. Booth,

however, further complements Ohms and Hazama by renaming and storing sorted dates in

the CUSTMER.DBF NEW CUST.DBF databases (p. 841). It would have been obvious to

one of ordinary skill in the art of data processing to combine the teachings of the cited

references. Booth's teaching of the storing the sorted reformatted dates would users of Ohms-

Hazama's system to readily retrieve the reformatted dates in their chronological sequence at any time

and in accordance with their needs.

49. Claim 63 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 63, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed 'method for representing and utilizing dates stored in at least one date field of a database

utilizing symbolic representations o f the dates stored in the at least one date field of the database,

which are in a format that creates ambiguity between dates in each of a pair of adjacent centuries' by

presenting a computer-implemented method for processing date outside the twentieth century (see

title, p 244 et seq). Ohms further discloses the steps of 'converting each of the symbolic

representations of dates stored in the at least one date field of the database to a symbolic

representation of each of the respective dates that does not create the ambiguity, by windowing the

symbolic representations of each of the respective dates as stored in the at least one date field of the

database against a pivot year represented by one of the symbolic representations of the dates as

stored in the at least one date field of' the database, without the addition of any new data field to the

database for purposes of such windowing and converting; and running a program collectively on the

stored converted symbolic representations to manipulate the dates represented by the converted

symbolic representations, separately from the symbolic representations of dates contained in the at

least one date field of the database' by detailing a short Gregorian format (MMDDYY) to represent

dates (p 247, see table 1), wherein said dates fall within a 10 decade or a 100 year window (p 249,

left-hand column, lines 3-7 et seq) and wherein the 100 year window contains dates that span in the

21st century (2000-2024) (p 248, right hand column). Ohms also indicates that years that are later or

equal (25-99) to the pivot date (25) fall within the 20th century thereby equating C1 C2 to 19 (i.e.

1925-1999), whereas dates that are earlier 900-24) than the pivot date fall within the 21st century

thereby equating C 1 C2 to 20 (i. e. 2000-2024) (see p 248, right-hand column. Additionally, Ohms

discloses that upon determining that a two-digit date falls within the 20th or the 21st the century, it is

expressed in accordance with its corresponding century without actually adding a new data field or

modifying the database (i.e. 25-99 -----> 1925-1999, and 00-24 -----> 2000-2024) (p 248, right hand

column).

Regarding the selection of a pivot year for the century window, Ohms discloses specifying a year as

the desired starting point (pivot date) of the 100 year window (p 248, right hand column, 2nd

paragraph). Ohms does not particularly detail that the pivot date is earlier than the earliest two digit

date in the database. Hazama, however, discloses an analogous date processing system wherein, for

a 100 year window, the pivot date for the window is selected based on a two digit year date that i s

smaller than the smallest two digit year date in the database (e.g. if smallest two digit data stored in

database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated

document, last paragraph). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. Hazama's teaching of the pivot date

being smaller than the smallest two digit date in the database would restrict the selection of Ohms's

window and thereby forcing all dates already stored in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of sorting the dates in the form C1

C2 YI Y2' However, Booth discloses an analogous system that utilizes the Clipper programming

language to process dates stored in a database to thereby derive other dates therefrom (p 939, lines

1-_> et seq. In particular, analogously to Ohms and Hazama, Booth discloses the SET EPOCH

command fir comparing an entered two digit date with the year digit (pivot date) of the epoch setting

to determine the century to place the date into (p 941, see SET EPOCH paragraph et seq) thereby

converting a six digit date (MMDDYY) into a corresponding eight digit date (C 1 C2Y 1 Y2M 1 M2D 1

D2) (see p. 940941). Additionally, Booth complements Ohms and Hazama by suggesting the sorting

of converted dates after having been reformatted by the STET EPOCH command (p. 945). It would

have been obvious to one of ordinary skill in the art of data processing at the time of the instant

invention to combine the teachings of the cited references. Booth's teaching of sorted reformatted

dates would facilitate the Ohms-Hazama's system to return the reformatted dates in chronological

sequence. And it would therefore be very useful when indexing the database in date order, as

suggested by Booth in page 945.

Ohms and Hazama do not specifically disclose the step of 'storing the converted

symbolic representations separate from the at least one date field of the database.' Booth,

however, further complements Ohms and Hazama by renaming and storing sorted dates in

the CUSTMER.DBF NEW CUST.DBF databases (p. X41). It would have been obvious to

one of ordinary skill in the art of data processing to combine the teachings of the cited

references. Booth's teaching of the storing the sorted reformatted dates would users of Ohms-

Hazama's system to readily retrieve the reformatted dates in their chronological sequence at any time

and in accordance with their needs. 50.

Claim 64 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of Hazama,

further in view of Booth

As to claim 64, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed 'method for representing and utilizing dates stored in at least one date field of a database

utilizing symbolic representations of the dates stored in the at least one date field of the database,

which are in a format that creates ambiguity between dates in each of a pair of adjacent centuries' by

presenting a computer-implemented method for processing date outside the twentieth century (see

title, p 244 et seq). Ohms further discloses the steps of 'converting each of the symbolic

representations of dates stored in the at least one date field of the database to a symbolic

representation of each of the respective dates that does not create the ambiguity, by windowing the

symbolic representations of each of the respective dates as stored in the at least one date field of the

database against a pivot year represented by one of the symbolic representations of the dates as

stored in the at least one date field of the database, without modifying any of the symbolic

representations of dates in the at least one date field of the database for purposes of such windowing

and converting; and running a program on the stored converted symbolic representations to

manipulate data in the database according to the dates represented by the converted symbolic

representations, separately from the symbolic representations of dates contained in the at least one

date field of the database' by detailing a short Gregorian format I,NINMDYY) to r--present dates (p

247, see table 1), wherein said dates fall within a 10 decade or a 100 year window (p 249, left-hand

column, lines 3-7 et seq) and wherein the 100 year window contains dates that span in the 21st

century (2000-2024) (p 248, right hand column). Ohms also indicates that years that are later or equal

(25-99) to the pivot date (25) fall within the 20th century thereby equating CIC2 to 19 (i.e. 1925-1999),

whereas dates that are earlier 900-24) than the pivot date fall within the 21st century thereby equating

CIC2 to 20 (i.e. 2000-2024) (see p 248, right-hand column. Additionally, Ohms discloses that upon

determining that a two-digit date falls within the 20th or the 21st the century, it is expressed in

accordance with its corresponding century without actually adding a new data field or modifying the

database (i.e. 25-99 -----> 1925-1999, and 00-24 -----> 2000-2024) (p 248, right hand column).

Regarding the selection of a pivot year for the century window, Ohms discloses

'specifying a year as the desired starting point (pivot date) of the 100 year window (p 248, right hand

column, 2nd paragraph). Ohms does not particularly detail that the pivot date is earlier than the

earliest two digit date in the database. Hazama, however, discloses an analogous date processing

system wherein, for a 100 year window, the pivot date for the window is selected based on a two digit

year date that is smaller than the smallest two digit year date in the database (e.g. if smallest two digit

date stored in database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of

translated document, last paragraph). It would have been obvious to one of ordinary skill in the art of

data processing to combine the teachings of the cited references. The ordinary skilled artisan having

read Ohms would immediately see the need to determine which 100 year span to use. This

determination would have led the ordinary skilled artisan to the Hazama reference, which teaches the

pivot date being smaller than the smallest two digit date in the database having all the dates within a

100 year period as a solution to restrict the selection of Ohms's window and thereby forcing all dates

already stored in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of sorting the dates in the form C1

C2 Y1 Y2' However, Booth discloses an analogous system that utilizes the Clipper programming

language to process dates stored in a database to thereby derive other dates therefrom (p 939, lines

1-3 et seq. In particular, analogously to Ohms and Hazama, Booth discloses the SET EPOCH

command for comparing an entered two digit date with the year digit (pivot date) of the epoch setting

to determine the century to place the date into (p 941, see SET EPOCH paragraph et seq) thereby

converting a six digit date (MMDDYY) into a corresponding eight digit date (C] C2YIY2MIM2DID2)

(see p. 940941). Additionally, Booth complements Ohms and Hazama by suggesting the sorting of

converted dates after having been reformatted by the SET EPOCH command (p. 945). It would have

been obvious to one of ordinary skill in the art of data processing at the time of the instant invention to

combine the teachings of the cited references. Booth's teaching of sorted reformatted dates would

facilitate the Ohms-Hazama's system to return the reformatted dates in chronological sequence. And

it would therefore be very useful when indexing the database in date order, as suggested by Booth in

page 945.

Ohms and Hazama do not specifically disclose the step of 'storing the converted

symbolic representations separate from the at least one date field of the database.' Booth,

however, further complements Ohms and Hazama by renaming and storing sorted dates in

the CUSTMER.DBF NEW CUST.DBF databases (p. 3341). It would have been obvious to

one of ordinary skill in the art of data processing to combine the teachings of the cited

references. Booth's teaching of the storing the sorted reformatted dates would users of Ohms-

Hazama's system to readily retrieve the reformatted dates in their chronological sequence at any time

and in accordance with their needs.

51. Claim 65 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 65, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed 'method for representing and utilizing dates stored in at least one date field of a database

utilizing symbolic representations of the dates stored in the at least one date field of the database,

which are in a format that creates ambiguity between dates in each of a pair of adjacent centuries' by

presenting a computer-implemented method for processing date outside the twentieth century (see

title, p 244 et seq). Ohms further discloses the steps of 'converting each of the symbolic

representations o F dates stored in the at least one date field of the database to a symbolic

representation of each of the respective dates that does not create the ambiguity, by windowing the

symbolic representations of each of the respective dates as stored in the at least one date field of the

database against a pivot year represented by one of the symbolic representations of the dates as

stored in the at least one date field of the database, without modifying any of the, symbolic

representations of dates in the at least one date field of the database for purposes of such windowing

and converting; and running a program collectively on the stored converted symbolic representations

to manipulate the dates represented by the converted symbolic representations, separately from the

symbolic representations of dates contained in the at least one date field of the database' by detailing

a short Gregorian format (MMDDYY) to represent dates (p 247, see table 1), wherein said dates fall

within a 10 decade or a 100 year window (p ';49, left-hand column, lines 3-7 et seq) and wherein the

100 year window contains dates that span in the 21st century (2000-2024) (p 248, right hand

column). Ohms also indicates that years that are later or equal (25-99) to the pivot date (25) fall within

the 20th century thereby equating C1 C2 to 19 (Le. 1925-1999), whereas dates that are earlier

900-24) than the pivot date fall within the 21st century thereby equating CIC2 to 20 (i.e. 2000-2024)

(see p 248, right-hand column. Additionally, Ohms discloses that upon determining that a two-digit

date falls within the 20th or the 21st the century, it is expressed in accordance with its corresponding

century without actually adding a new data field or modifying the database (Le. 25-99 ------>

1925-1999, and 00-24 -----> 2000-2024) (p 248, right hand column).

Regarding the selection of a pivot year for the century window, Ohms discloses specifying a year as

the desired starting point (pivot date) of the 100 year window (p 248, right hand column, 2nd

paragraph). Ohms does not particularly detail that the pivot date is earlier than the earliest two digit

date in the database. Hazama, however, discloses an analogous date processing system wherein, for

a 100 year window, the pivot date for the window is selected based on a two digit year date that is

smaller than the smallest two digit year date in the database (e.g. if smallest two digit date stored in

database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated

document, last paragraph). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited :references. The ordinary skilled artisan having read

Ohms would immediately see the need to determine which 100 year span to use. This

determination would have led the ordinary skilled artisan to the Hazama reference, which

teaches the pivot date being smaller than the smallest two digit date in the database having all the

dates within a 100 year period as a solution to restrict the selection of Ohms's window and thereby

forcing all dates already stored in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of sorting the dates in the form C1

C2 YI Y2' However, Booth discloses an analogous system that utilizes the Clipper programming

language to process dates stored in a database to thereby derive other dates therefrom (p 939, lines

1-3 et seq. In particular, analogously to Ohms and Hazama, Booth discloses the SET EPOCH

command for comparing an entered two digit date with the year digit (pivot date) of the epoch setting

to determine the century to place the date into (p 941, see SET EPOCH paragraph et seq) thereby

converting a six digit date (MMDDYY) into a corresponding eight digit date (C I C2YI Y2M I N42D 1

D2) (see p. 940941). Additionally, Booth complements Ohms and Hazama by suggesting the sorting

of converted dates after having been reformatted by the SET EPOCH command (p. 945). It would

have been obvious to one of ordinary skill in the art of data processing at the time of the instant

invention to combine the teachings of the cited references. Booth's teaching of sorted reformatted

dates would facilitate the Ohms-Haz,3,ma's system to return the reformatted dates in chronological

sequence. And it would therefore be very useful when indexing the database in date order, as

suggested by Booth in page 945.

Ohms and Hazama do not specifically disclose the step of 'storing the converted symbolic

representations separate from the at least one date field of the database.' Booth, however, further

complements Ohms and Hazama by renaming and storing sorted dates in the CUSTMER.DBF NEW

CUST.DBF databases (p. 841). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. Booth's teaching of the storing the

sorted reformatted dates would users of OhmsHazama's system to readily retrieve the reformatted

dates in their chronological sequence at any time and in accordance with their needs.

52. Claim 66 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 66, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed 'method of processing dates in a database' by presenting a computer implemented method

for processing date outside the twentieth century (see title, p 244 et seq). Ohms further discloses the

step of 'providing a database with dates stored in at least one date field therein according to a

17ormat wherein M:1 M2 is the numerical month designator, D1 D2 is the numerical day designator,

and Y1 Y2 is the numerical year designator' by detailing a short Gregorian format (MMDDYY) to

represent dates (p 247, see table 1), wherein said dates fall within a 10 decade or a 100 year window

(p 249, left-hand column, lines 3-7 et seq) and wherein the 100 year window contains dates that span

in the 21st century (2000-2024) (p 248, right hand column). Ohms also discloses the step of

'determining a century designator C1 IJ2 for each date in the database, C1 C2 having A first value if

Y1 Y2 is less than YAYB and having a second value if YI Y2 is equal to or greater than YAYB' by

indicating that years that are later or equal (25-99) to the pivot date (25) fall within the 20th century

thereby equating CIC2 to 19 (i.e. 1925-1999), whereas dates that are earlier 900-24) than the pivot

date fall within the 21st century thereby equating C1 C2 to 20 (Le. 2000-2024) (see p 248, right-hand

column. Additionally, Ohms discloses the step of 'reformatting the symbolic representation of each

symbolic representation of a date in a portion of the at least one date field in the database, without

the addition of any new data field to the database, with the reformatted symbolic representation of

Application/Control Number● 9/512,592; 90/005,592; 90/005,62● 0/005727;   Page
90/006,541   161
Art Unit: 2161

each date in the database having the values C 1 C2, Y 1 Y2, M 1 M2, and D 1 D2' by indicating that upon determining that a two-digit date falls within the 20th or the 21st the century, it is expressed in accordance with its corresponding century without actually adding a new data field or modifying the database (i.e. 25-99 -----> 1925-1999, arid 00-24 -----> 2000-2024) (p 248, right hand column). Regarding the step of' selecting a window with YAYB value for A pivot date of the window, YAYB being no later than the earliest Y1 Y2 year designator in the database,' Ohms discloses specifying a year as the desired starting point (pivot date) of the 100 year window (p 248, right hand column, 2nd paragraph). Ohms does not particularly detail that the pivot date is earlier than the earliest two digit date in the database. Hazama, however, discloses an analogous date processing system wherein, for a 100 year window, the pivot date for the window is selected based on a two digit year date that is smaller than the smallest two digit year date in the database (e.g. if smallest two digit date stored in database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated document, last paragraph). It would have been obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited references. Hazama's teaching of the pivot date being smaller than the smallest two digit date in the database would restrict the selection of Ohms's window and thereby forcing all dates already stored in the database to fall in the 20th century. Further, Ohms and Hazama do not specifically, disclose the step of 'repeating the step of reformatting until each symbolic representation of a date in the at least one date field has been reformatted in order to facilitate collectively further processing the reformatted symbolic representations of each of the symbolic representations of each of the dates.' However, Booth discloses an analogous system that utilizes the Clipper programming language to process dates stored in a database to thereby derive other dates therefrom (p 939, lines 1-3 et seq. In particular, analogously to Ohms and Hazama, Booth discloses the SET EPOCH command for comparing an entered two digit date with the year

digit (pivot date) of the epoch setting to determine the century to place the date into (p 941, see SET

EPOCH paragraph et seq) thereby converting a six digit date (AZMDDYY) into a corresponding eight

digit date (CIC2YIY2MIM2DID2) (see p. 940-941). Additionally, Booth complements Ohms and

Hazama by suggesting the sorting of converted dates after having been reformatted by the SET

EPOCH column and (p. 945). It would have been obvious to one of ordinary skill in the art of data

processing at the time of the instant invention to combine the teachings of the cited references.

Booth's teaching of sorted reformatted dates would facilitate the Ohms-Hazama's system to return the

reformatted dates in chronological sequence. And it would therefore be very useful when indexing the

database in date order, as suggested by Booth in page 945.

53. Claim 67 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 67, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed 'method of processing dates in a database' by presenting a computer implemented method

for processing date outside the twentieth century (see title, p 244 et seq). Ohms further discloses the

step of ' providing a database with dates stored in at least one date field therein according to A format

wherein Y I Y2 is the numerical year designator' by detailing a short Gregorian format (MMDDYY) to

represent dates (p 247, see table 1), wherein said dates fall within a 10 decade or a 100 year window

(p 249, left-hand column, lines 3-7 et seq) and wherein the 100 year window contains dates that span

in the 21st century (2000-2024) (p 248, right hand column). Ohms also discloses the step of

'determining a century designator C1 C2 for each date in the database, C1 C2 having A first value if

Y1 Y2 is less than YAYB and having a second value if Y1 Y2 is equal to or greater than YAYB' by

indicating that years that are later or equal (25-99) to the pivot date (25) fall within the 20th century

thereby equating C I C2 to 19 (L e. I 925-1999), whereas dates that are earlier 900-24) than the pivot

date fall within the 21st century thereby equating C 1 C2 to 20 (i.e. 2000-2024) (see p 248, right-hand

column. Additionally, Ohms discloses the step of `reformatting the symbolic representation of each

symbolic representation of a date in a portion of the at least one date field in the database, without

the addition of any new data field to the database, with the reformatted symbolic representation of

each date in the database having the values C 1 C2, Y 1 Y2' by indicating that upon determining that

a two digit date falls within the 20th or the 21st the century, it is expressed in accordance with its

corresponding century without actually adding a new data field or modifying the database (i.e. 25-99

-----> 1925-1999, and 00-24 ----> 2000-2024) (p 248, right hand column). ΄

Regarding the step of `selecting a window with a YAYB value for a pivot date of the window, YAYB

being no later than the earliest Y 1 Y2 year designator in the database,' Ohms discloses specifying a

year as the desired starting point (pivot date) of the 100 year window (p 248, right hand column, 2nd

paragraph). Ohms does not particularly detail that the pivot date is earlier than the earliest two digit

date in the database. Hazama, however, discloses an analogous date processing system wherein, for

a 100 year window, the pivot date for the window is selected based on a two digit year date that is

smaller than the smallest two digit year date in the database (e.g. if smallest two digit date stored in

database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated

document, last paragraph). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. The ordinary skilled artisan having read

Ohms would immediately see the need to determine which 100 year span to use. This determination

would have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date

being smaller than the smallest two digit date in the database having all the dates within a 100 year

period as a solution to restrict the selection of Ohms's window and thereby forcing all dates already

stored in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of 'repeating the step of reformatting

until each symbolic representation of a date in the at least one date field has been reformatted in

order to facilitate collectively further processing the reformatted symbolic representations of each of

the symbolic representations of each of the dates.' However, Booth discloses an analogous system

that utilizes the Clipper programming language to process dates stored in a database to thereby

derive other dates therefrom (p 939, lines 1-3 et seq. In particular, analogously to Ohms and Hazama,

Booth discloses the SET EPOCH command for comparing an entered two digit date with the year

digit (pivot date) of the epoch setting to determine the century to place the date into (p 941, see SET

EPOCH paragraph et seq) thereby converting a six digit date (NTVIDDYY) into a corresponding eight

digit date (C 1 C2Y 1 Y2M I M2D 1 D2) (see p. 940-941). Additionally, Booth complements Ohms and

Hazama by suggesting the sorting of converted dates after having been reformatted by the SET

EPOCH command (p. 945). It would have been obvious to one of ordinary skill in the art of data

processing at the time of the instant invention to combine the teachings of the cited references.

Booth's teaching of sorted reformatted dates would facilitate the Ohms-Hazama's system to return the

reformatted dates in chronological sequence. And it would therefore be very useful when indexing the

database in date order, as suggested by Booth in page 945.

54. Claim 68 is rejected under 35 U.S. C. 103(a) as being unpatentable over Ohms in view of

Hazama.

As to claim 68, Ohms substantially discloses the claimed invention. In particular, Ohms

discloses the claimed 'method of processing symbolic representations of dates stored in a database'

by presenting a computer-implemented method for processing date outside the twentieth century (see

title, p 244 et seq). Ohms further discloses the step of 'providing a database with symbolic

representations of dates stored in at least one date field therein according to a format wherein YI Y2

is the numerical year designator' by detailing a short Gregorian format (MMDDYY) to represent dates

(p 247, see table 1), wherein said dates fall within a 10 decade or a 100 year window (p 249,

left-hand column, lines 3-7 et seq) and wherein the 100 year window contains dates that span in the

21st century (2000-2024) (p 248, right hand column). Ohms also discloses the step of 'determining a

century designator C1 C2 for each symbolic representation of a date in the database, C1 C2 having a

first value if Y 1 Y2 is less than YAYB and having a second value if YI Y2 is equal to or greater than

YAYB' by indicating that years that are later or equal (25-99) to the pivot date (25) fall within the 20th

century thereby equating C1 C2 to 19 (Le. 1925-1999), whereas dates that are earlier 900-24) than

the pivot date fall within the 21st century thereby equating C 1 C2 to 20 (i.e. 2000-2024) (see p 248,

right-hand column. Additionally, Ohms discloses the step of 'reformatting the symbolic representation

of each symbolic representation of a date in at least one date field in the database, without the

addition of any new data field to the database, with the reformatted symbolic representation of each

date in the database having the values C 1 C2, Y 1 Y2, in order to facilitate further processing of the

reformatted symbolic representations of each of the symbolic representations of each of the dates, by

running a program on the reformatted symbolic ;representations or each of the dates' by indicating

that upon determining that a two-digit date falls within the 20th or the 21st the century, it is expressed

in accordance with its corresponding century without actually adding a new data field or modifying the

database (i.e. 2-`i-99 ----> 1925-1999, and 00-24 ----> 2000-2024) (p 248, right hand column).

Regarding the step of `selecting a window with a YAYB value for the first decade of the window,

YAYB being no later than the earliest Y I Y2 year designator in the at least one date field of the

database,' Ohms discloses specifying a year as the desired starting point (pivot date) of the 100 year

window (p 248, right hand column, 2nd paragraph). Ohms does not particularly detail that the pivot

date is earlier than the earliest two digit date in the database. Hazama, however, discloses an

analogous date processing system wherein, for a 100 year window, the pivot date for the window is

selected based on a two digit year date that is smaller than the smallest two digit year date in the

database (e.g. if smallest two digit date stored in database is 73, the pivot date for the 100 year

window is chosen to be 72). (page 4 of translated document, last paragraph). It would have been

obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited

references. The ordinary skilled artisan having read Ohms would immediately see the need to

determine which 100 year span to use. This determination would have led the ordinary skilled artisan

to the Hazama reference, which teaches the pivot date being smaller than the smallest two digit date

in the database having all the dates within a 100 year period as a solution to restrict the selection of

Ohms's window and thereby forcing all dates already stored in the database to fall in the 20th century.

55.   Claim 69 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

       As to claim 69, Ohms substantially discloses the claimed invention. In particular, Ohms

discloses the claimed 'method of processing dates in a database' by presenting a computer -

implemented method for processing date outside the twentieth century (see title, p 244 et seq). Ohms

further discloses the step ~of 'providing a database with dates stored in at least one date field therein

according to a format wherein YI Y2 is the numerical year designator' by detailing a short Gregorian

format (MMDDYY) to represent dates (p 247, see table 1), wherein said dates fall within a 10 decade

or a 100 year window (p 249, left-hand column, lines 3-7 et seq) and wherein the 100 year window

contains dates that span in the 21st century (2000-2024) (p 248, right hand column). Ohms also

discloses the step of 'determining a century designator CI (I2 for each date in the at least one date

field of the database, CI C2 having a first value if YI Y2 is less than YAYB and having a second value

if YI Y2 is equal to or greater than YAYB' by indicating that years that are later or equal (25-99) to the

pivot date (25) fall with in the 20th century thereby equating C 1 C2 to 19 (i. e. 1925-1999), whereas

dates that are earlier 900-24) than the pivot date fall within the 21st century thereby equating C 1 C2

to 20 (i. e. 2000-2024) (see p 248, right-hand column. Additionally, Ohms discloses the step of

reformatting the symbolic representation of each symbolic representation of a date in the at least one

date field in the database, without the addition of any new data field to the database, with the;

reformatted symbolic representation of each date in the database having the a values C 1 C2, Y 1 Y2'

by indicating that upon determining that a two-digit date falls within the 20th or the 21st the century, it

is expressed in accordance with its corresponding century without actually adding a new data field or

modifying the database (i.e. 25-99 -----> 1925-1999, and 00-24 -----> 2000-2024) (p 248, right hand

column).

Regarding the step of 'selecting a window with a YAYB value for the first decade of the window,

YAYB being no later than the earliest Y 1 Y2 year designator in the at least one date field of the

database,' Ohms discloses specifying a year as the desired starting point (pivot date) of the 100 year

window (1) 248, right hand column, 2nd paragraph). Ohms does not particularly detail that the pivot

date is earlier than the earliest two digit date in the database. Hazama, however, discloses an

analogous dare processing system wherein, for a 100 year window, the pivot date for the window is

selected based on a two digit year date that is smaller than the smallest two digit year date in the

database (e.g. if smallest two digit date stored in database is 73, the pivot date for the 100 :year

window is chosen to be 72). (page 4 of translated document, last paragraph). It would have been

obvious to one of ordinary skill in the art of data processing to combine the teachings of the cited

references. Hazama's teaching of the pivot date being smaller than the smallest two digit date in the

database would restrict the selection of Ohms's window and thereby forcing all dates already stored

in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of 'sorting the

reformatted symbolic representations of the dates in the form CI C2 Y1 Y2; and running a

program on the reformatted symbolic representations o E each of the dates' However, Booth

discloses an analogous system that utilizes the Clipper programming language to process

dates stored in a database to thereby derive other dates therefrom (p 939, lines 1-3 et seq. In

particular, analogously to Ohms and Hazama, Booth discloses the SET EPOCH command for

comparing an entered two digit date with the year digit (pivot date) of the epoch setting to determine

the century to place the date into (p 941, see SET EPOCH paragraph et seq) thereby converting a six

digit date (MMIDDYY) into a corresponding eight digit date (CIC2Y1Y2MIM2DID2) (seep. 940--941).

Additionally, Booth complements Ohms and Hazama by suggesting the sorting of converted dates

after having been reformatted by the SET EPOCH command (p. 945). It would have been obvious to

one of ordinary skill in the art of data processing at the time of the instant invention to combine the

teachings of the cited references. Booth's teaching of sorted reformatted dates would facilitate the

Ohms Hazama's system to return the reformatted dates in chronological sequence. And, it would

therefore be very useful when indexing the database in (late order, as suggested by Booth in page

945.

56.   Claim 70 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 70, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed `method for representing and utilizing dates stored in at least one date field of a database

utilizing symbolic representations of the dates stored in at least one date field of the database, which

are in a format that creates ambiguity between dates in each of a pair of adjacent centuries' by

presenting a computer-implemented method for processing date outside the twentieth century (see

title, p 244 et seq). Ohms further discloses the steps of `converting each of the symbolic

representations of dates stored in the at least one date field of the database to a symbolic

representation of each of the respective dates that does not create the ambiguity, by windowing the

symbolic representations of each of the respective dates as stored in the at least one date field of the

database against a pivot year represented by one of the symbolic representations of the dates as

stored in the at least one date field of the database, without modifying any of the symbolic

representations of dates in the at least one date field of the database for purposes of such windowing

and converting; and running a program collectively on the stored converted symbolic representations

to manipulate the dates represented by the converted symbolic representations, separately from the

symbolic representations of dates contained in the at least one date field of the database' by detailing

a short Gregorian format (MMDDYY) to represent dates (p 247, see table 1), wherein said dates fall

within a 10 decade or a 100 year window (p 2-49, left-hand column, lines 3-7 et seq) and wherein the

100 year window contains dates that span in the 21st century (2000-2024) (p 248, right hand

column). Ohms also indicates that years that are later or equal (25-99) to the pivot date (25) fall within

the 20th century thereby equating C1 C2 to 19 (Le. 1925-1999), whereas dates that are earlier

900-24) than the pivot date fall within the 21st century thereby equating CIC2 to 20 (i.e. 2000-2024)

(see p 248, right-hand column. Additionally, Ohms discloses that upon determining that a two-digit

date falls within the 20th or the 21st the century, it is expressed in accordance with its corresponding

century without actually adding a new data field or modifying the database (i.e. 25-99 ----->

1925-1999, and 00-24 ----> 2000-2024) (p 248, right hand column).

Regarding the selection of a pivot year for the century window, Ohms discloses specifying a year as

the desired starting point (pivot date) of the 100 year window (p 248, right hand column, 2nd

paragraph). Ohms does not particularly detail that the pivot date is earlier than the earliest two digit

date in the database. Hazama, however, discloses an analogous date processing system wherein, for

a 100 year window, the pivot date for the window is selected based on a two digit year date that is

smaller than the smallest two digit year date in the database (e.g. if smallest two digit date stored in

database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated

document, last paragraph). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. The ordinary skilled artisan having read

Ohms would immediately see the need to determine which 100 year span to use. This determination

would have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date

being smaller than the smallest two digit date in the database having all the dates within a 100 year

period as a solution to restrict the selection of Ohms's window and thereby forcing all dates already

stored in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of sorting the dates in the form CI

C2 Y1 Y2' However, Booth discloses an analogous system that utilizes the Clipper programming

language to process dates stored in a database to thereby derive other dates therefrom (p 939, lines

1-3 et seq. In particular, analogously to Ohms and Hazama, Booth discloses the SET EPOCH

command for comparing an entered two digit date with the year digit (pivot date) of the epoch setting

to determine the century to place the date into (p 941, see SET EPOCH paragraph et seq) thereby

converting a six digit date (MMDDYY) into a corresponding eight digit date (C IC2Y 1 Y2M 1 M2D I

D2) (see p. 940941). Additionally, Booth complements Ohms and Hazama by suggesting the sorting

of converted dates after having been reformatted by the SET EPOCH command (p. 945). It would

have been obvious to one of ordinary skill in the art of data processing at the time of the instant

invention to combine the teachings of the cited references. Booth's teaching of sorted reformatted

dates would facilitate the Ohms-Hazama's system to return the reformatted dates in chronological

sequence. And it would therefore be very useful when indexing the database in date order, as

suggested by Booth in page 945.

Ohms and Hazama do not specifically disclose the step of 'storing the converted symbolic

representations separate from the at least one date field of the database. Booth, however, further

complements Ohms and Hazama by renaming and storing sorted dates in the CUSTMERDBF NEW

CUST.DB.F databases (p. 841). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. Booth's teaching of the storing the

sorted reformatted dates would users of Ohms-Hazama's system to readily retrieve the reformatted

dates in their chronological sequence at any time and in accordance with their needs.

57.   Claim 71 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 71, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed 'method for representing and utilizing dates stored in at least one date field of the database

utilizing symbolic representations of the dates stored in the at least one date field of the database,

which are in a format that creates ambiguity between dates in each of a pair of adjacent centuries' by

presenting a computer-implemented method for processing date outside the twentieth century (see

title, p 244 et seq). Ohms further discloses the steps of 'converting each of the symbolic

representations of dates stored in the at least one date field of the database to a symbolic

representation of each of the respective dates that does not create the ambiguity, by windowing the

symbolic representations of each of the respective dates as stored in the at least one date field of the

database against a pivot year, with the pivot year being less than or equal to the earliest date

represented by a symbolic representation of dates stored in the at least one date field, and without

the addition of any new data field to the database for purposes of such windowing and converting;

and, running a program on the stored converted symbolic representations of each of the converted

symbolic representations of the dates to sort or otherwise manipulate the dates represented by the

converted symbolic representations, separately from the dare data symbolic representations

contained in the at least one date field of the database' by detailing a short Gregorian format

(MMDDYY) to represent dates (p 247, see table 1), wherein said dates fall within a 10 decade or a

100 year window (p 249, left-hand column, lines 3-7 et seq) and wherein the 100 year window

contains dates that span in the 21st century (2000-2024) (p 248, right hand column). Ohms

also indicates that years that are later or equal (25-99) to the pivot date (25) fall within the 20th

century thereby equating C1 C2 to 19 (Le. 1925-1999), whereas dates that are earlier 900-24) than

the pivot date fall within the 21st century thereby equating C I C2 to 20 (i. e. 2000-2024) (see p 248,

right-hand column. Additionally, Ohms discloses that upon determining that a two-digit date falls

within the 20th or the 21st the century, it is expressed in accordance with its corresponding century

without actually adding A new data field or modifying the database (i.e. 25-99 -----> 1925-1999, and

00-24 -----> 2000-2024) (p 248, right hand column).

Regarding the selection of a pivot year for the century window, Ohms discloses specifying a year as

the desired starting point (pivot date) of the 100 year window (p 248, right hand column, 2nd

paragraph). Ohms does not particularly detail that the pivot date is earlier than the earliest two digit

date in the database. Hazama, however, discloses an analogous date processing system wherein, for

a 100 year window, the pivot date for the window is selected based on a two digit year date that is

smaller than the smallest two digit year date in the database (e.g. if smallest two digit date stored in

database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated

document, last paragraph). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. The ordinary skilled artisan having read

Ohms would immediately see the need to determine which 100 year span to use. This determination

would have led the ordinary skilled artisan to the Hazama reference, which teaches the pivot date

being smaller than the smallest two digit date in the database having all the dates within a 100 year

period as a solution to restrict the selection of Ohms's window and thereby forcing all dates already

stored in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of sorting the

dates in the form C1 C2 Y1 Y2' However, Booth discloses an analogous system that utilizes

the Clipper programming language to process dates stored in a database to thereby derive

other dates therefrom (p 939, lines 1-3 et seq. In particular, analogously to Ohms and

Hazama, Booth discloses the SET EPOCH command for comparing an entered two digit date with

the year digit (pivot date) of the epoch setting to determine the century to place the date into (p 941,

see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a

corresponding eight digit date (C I C2Y 1 Y2M 1 N12D 1 D2) (see p. 940941). Additionally, Booth

complements Ohms and Hazama by suggesting the sorting of converted dates after having been

reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of ordinary

skill in the art of data processing at the time of the instant invention to combine the teachings of the

cited references. Booth's teaching of sorted reformatted dates would facilitate the Ohms-Hazama's

system to return the reformatted dates in chronological sequence. And it would therefore be very

useful when indexing the database in date order, as suggested by Booth in page 945.

Ohms and Hazama do not specifically disclose the step of `storing the converted symbolic

representations separate from the at least one &ate field of the database.' Booth, however, further

complements Ohms and Hazama by renaming and storing sorted dates in the CUSTMER.DBF NEW

CUST.DBF databases (p. 841). It would have been obvious to one of ordinary skill in the art of data

processing to com0ine the teachings of the cited references. Booth's teaching of the storing the

sorted reformatted dates would users of Ohms-Hazama's system to readily retrieve the reformatted

dates in their chronological sequence at any time and in accordance with their needs.

58. Claim 72 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama.

As to claim 72, Ohms substantially discloses the claimed invention. In particular, Ohms discloses the

claimed 'method of processing symbolic representations of dates stored in a database' by presenting

a computer-implemented method for processing date outside the twentieth century (see title, p 244 et

seq). Ohms further discloses the step of 'selecting a database with symbolic representations of dates

stored therein according to a format wherein MI M2 is the numerical month designator, DI D2 is the

numerical day designator, and YI Y2 is the numerical year designator' by detailing a short Gregorian

format (MMDDYY) to represent dates (p 247, see table 1), wherein said dates fall within a 10 decade

or a 100 year window (p 249, left-hand column, lines 3-7 et seq) and wherein the 100 year window

contains dates that span in the 21st century (2000-2024) (p 248, right hand column). Ohms also

discloses the step of 'determining ,a century designator CI C2 for each symbolic representation of a

date in the database, C1 C2 having a first value if Y1 Y2 is less than YAYB and having a second

value if Y I Y2 is equal to or greater than YAYB' by indicating that years that are later or equal (25-99)

to the pivot date, (25) fall within the 20th century thereby equating C 1 C2 to 19 (L e. 1925-1999),

whereas (sates that are earlier 900-24) than the pivot date fall within the 21st century thereby

equating; CIC2 to 20 (i.e. 2000-2024) (see p 248, right-hand column. Additionally, Ohms discloses

the step of `reformatting the symbolic representation of each symbolic representation of a date in the

database with the values C1 C2, Y1 Y2, MI M2, and DI D2 prior to collectively further processing

information contained within the database associated with the respective dates' by indicating that

upon determining that a two-digit date falls within the 20th or the 21st the century, it is expressed in

accordance with its corresponding century without actually adding a new data field or modifying the

database (i.e. 25-99 ------> 1925-1999, and 00-24 ----> 2000-2024) (p 248, right hand column).

Regarding the step of `selecting a 10-decade window with a YAYB value for the first decade of the

window, YAYB being no later than the earliest YIY2 year designator in the database,' Ohms discloses

specifying a year as the desired starting point (pivot date) of the 100 year window (p 248, right hand

column, 2nd paragraph). Ohms does not particularly detail that the pivot date is earlier than the

earliest two digit date in the database. Hazama, however, discloses an analogous date; processing

system wherein, for a 100 year window, the pivot date for the window is selected based on a two digit

year date that is smaller than the smallest two digit year date in the database (e.g. if smallest two digit

date stored in database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of

translated document, last paragraph). It would have been obvious to one of ordinary skill in the art of

data processing to combine the teachings of the cited references. The ordinary skilled artisan having

read Ohms would immediately see the need to determine which 100-year span to use. This

determination would have led the ordinary skilled artisan to the Hazama reference, which teaches the

pivot date being smaller than the smallest two digit date in the database having all the dates within a

100 year period as a solution to restrict the selection of Ohms's window and thereby forcing all dates

already stored in the database to fall in the 20th century.

59.   Claim 73 is rejected under 35 U.S.C_ 103(a) as being unpatentable over Ohms in view of

Hazama.

As to claim 73, Ohms substantially discloses the claimed invention. In particular, Ohms

discloses the claimed 'method of processing symbolic representations of dates stored in a database'

by presenting a computer-implemented method for processing date outside the twentieth century (see

title, p 244 et seq). Ohms further discloses the step of 'providing a database with symbolic

representations of dates stored therein according to a format wherein YI Y2 is the numerical year

designator, all of the symbolic representations of dates falling within a 10-decade period of time' by

detailing a short Gregorian format (MMDDYY) to represent dates (p 247, see table 1), wherein said

dates fall within a 10 decade or a 100 year window (p 249, left-hand column, lines 3-7 et seq) and

wherein the 100 year window contains dates that span in the 21st century (2000-2024) (p 248, right

hand column). Ohms also discloses the step of 'determining a century designator CI C2 for each

symbolic representation of a date in the database, C1 C2 having a first value if YI Y2 is less than a

and having a second value if Y1 Y2 equal to or greater than YAYB' by indicating that years that are

later or equal (25-99) to the pivot date (25) fall within the 20th century thereby equating CIC2 to 19

(i.e. 1925-1999):, whereas dates that are earlier 900-24) than the pivot date fall within the 21st

century thereby equating CI C:2 to 20 (i.e. 2000-2024) (see p 248, right-hand column. Additionally,

Ohms discloses the step of `reformatting the symbolic representation of the date with the values CI

C2, YI Y:2, to facilitate further processing of the dates' by indicating that upon determining that a

two-digit date falls within the 20th or the 21st the century, it is expressed in accordance with its

corresponding century without actually adding a new data field or modifying the database (i.e. 25-99

-----> 1925-1999, and 00-24 ----> 2000-2024) (p 248, right hand column).

Regarding the step of 'selecting a 10-decade window with a YAYB value for the first decade of the

window, YAYB being no later than the earliest Y1 Y2 year designator in the database,' Ohms

discloses specifying a year as the desired starting point (pivot date) of the 100 year window (p 248,

right hand column, 2nd paragraph). Ohms does not particularly detail that the pivot date is earlier than

the earliest two digit date in the database. Hazama, however, discloses an analogous date

processing system wherein, for a 100 year window, the pivot date for the window is selected based

on a two digit year date that is smaller than the smallest two digit year date in the database (e.g. if

smallest two digit date stored in database is 73, the pivot date for the 100 year window is chosen to

be 72). (page 4 of translated document, last paragraph). It would have been obvious to one of

ordinary skill in the art of data processing to combine the teachings of the cited r6erences. The

ordinary skilled artisan having read Ohms would immediately see the need to determine which 100

year span to use. This determination would have led the ordinary skilled artisan to the Hazama

reference, which teaches the pivot date being smaller than the smallest two digit date in the database

having all the dates within a 100 year period as a solution to restrict the selection of Ohms's window

and thereby forcing all dates already stored in the database to fall in the 20th century.

60.   Claim 74 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 74, Ohms substantially discloses the claimed invention. In particular, Ohms

discloses the claimed 'method of processing dates in a database' by presenting a computer

implemented method for processing date outside the twentieth century (see title, p 244 et seq). Ohms

further discloses the step of 'providing a database with symbolic representations of dates stored

therein according to a format wherein YI Y2 is the numerical year designator, all of symbolic

representations of dates falling within a 10-decade period of time' by detailing a short Gregorian

format (M'VIDDYY) to represent dates (p 247, see table 1), wherein said dates fall within a 10 decade

or a 100 year window (p 249, left-hand column, lines 3-7 et seq) and wherein the 100 year window

contains dates that span in the 21st century (2000-2024) (p 248, right hand column). Ohms; also

discloses the step of 'determining a century designator C1 C2 for each date in the database, C1 C2

having a first value if Y1 Y2 is less than YAYB and having a second value if Y1 Y2 is equal to or

greater than YAYB' by indicating that years that are later or equal (25-99) to the pivot date (25) fall

within the 20th century thereby equating C1 C2 to 19 (i. e. 1925-1999), whereas dates that are earlier

900-24) than the pivot date fall within the 21st century thereby equating C 1 C2 to 20 (i.e. 2000-2024)

(see p 248, right-hand column. Additionally, Ohms discloses the step of reformatting each date in the

form C1 C2 YI Y2 to facilitate further processing of the dates' by indicating that upon determining that

a two-digit date falls within the 20th or the 21st the century, it is expressed in accordance with its

corresponding century without actually adding a new data field or modifying the database (i.e. 25-99

-----> 1925-1999, and 00-24 -----> 2000-2024) (p 248, right hand column).

Regarding the step of 'selecting a 10-decade window with a YAYB value for the first decade of the

window, YAYB being no later than the earliest Y 1 Y2 year designator in the database,' Ohms

discloses specifying a year as the desired starting point (pivot date) of the 100 year window (p 248,

right hand column, 2nd paragraph). Ohms does not particularly detail that the pivot date is earlier than

the earliest two digit date in the database. Hazama, however, discloses an analogous date

processing system wherein, for a 100 year window, the pivot date for the window is selected based

on a two digit year date that is smaller than the smallest two digit year date in the database (e.g. if

smallest two digit date stored in database is 73, the pivot date for the 100 year window is chosen to

be 72). (page 4 of translated document, last paragraph). It would have been obvious to one of

ordinary skill in the art of data processing to combine the teachings of the cited references. Hazama's

teaching of the pivot date being smaller than the smallest two digit date: in the database would restrict

the selection of Ohms's window and thereby forcing all dates already stored in the database to fall in

the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of sorting the dates in the form CI

C2 Y1 Y2.' However, Booth discloses an analogous system that utilizes the Clipper programming

language to process dates stored in A database to thereby derive other dates therefrom (p 939, lines

1-3 et seq. In particular, analogously to Ohms and Hazama, Booth discloses the SET EPOCH

command for comparing an entered two digit date with the year digit (pivot date) of the epoch setting

to determine the century to place the date into (p 941, see SET EPOCH paragraph et sec) thereby

converting a six digit date (MMDDYY) into a corresponding eight digit date (CI C2Y 1 Y2M 1 M2D 1

D2) (see p. 940941). Additionally, Booth complements Ohms and Hazama by suggesting the sorting

of converted dates after having been reformatted by the SET EPOCH command (p. 945). It would

have been obvious to one of ordinary skill in the art of data processing at the time of the instant

invention to combine the teachings of the cited references. Booth's teaching of sorted reformatted

dates would facilitate the Ohms -Hazama's system to return the reformatted dates in chronological

sequence. And, it would therefore be very useful when indexing the database in date order, as

suggested by Booth in page 945.

61.   Claim 75 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 75, Ohms substantially discloses the claimed invention. In particular, Ohms

discloses the claimed 'method of processing symbolic representations of dates stored in a database'

by presenting a computer-implemented method for processing date outside the twentieth century (see

title, p 244 et seq). Ohms further discloses the step of 'providing a database with symbolic

representations of dates stored therein according to a format wherein M I M2 is the numerical month

designator, D 1 D2 is the numerical day designator, and Y I Y2 is the numerical year designator' by

detailing a short Gregorian format (MMDDYY) to represent dates (p 247, see table 1), wherein said

dates fall within a 10 decade or a 100 year window (p 249, left-hand column, lines 3-7 et seq) and

wherein the 100 year window contains dates that span in the 21st century (2000-2024) (p 248, right

hand column). Olin is also discloses the step of 'determining a century designator CI C2 for each

symbolic representation of a date in the database, C1 C2 having a first value if YI Y2 is less than

YAYB and having a second value if Y 1 Y2 is equal to or greater than YAYB' by indicating that years

that are later or equal (25-99) to the pivot date (25) fall within the 20th century thereby equating CIC2

to 19 (i.e. 1925-1999), whereas dates that are earlier 900-24) than the pivot date fall within the 21st

century thereby equating C 1 C2 to 20 (i. e. 2000-2024) (see p 248, right-hand column. Additionally,

Ohms discloses the step of 'reformatting the symbolic representation of each symbolic representation

of a date in the database, without the addition of any new data field to the database, with the

reformatted symbolic representation of each date in the database having the values C 1 C2, Y 1 Y2,

MIM2, and D I D2 in order to facilitate further processing of the reformatted symbolic representations

of each of the symbolic representations of each of the dates' by indicating that upon determining that

a two digit date falls within the 20th or the 21st the century, it is expressed in accordance with its

corresponding century without actually adding a new data field or modifying the database (i.e. 25-99

-----> 1925-1999, and 00-24 ----> 2000-2024) (p 248, right hand column).

Regarding the step of 'selecting a window with a YAYB value for a pivot date of the window, YAYB

being no later than the YAYB earliest Y 1 Y2 year designator in the database,' Ohms discloses

specifying ;a year as the desired starting point (pivot date) of the 100 year window (p 248, right hand

column, 2nd paragraph). Ohms does not particularly detail that the pivot date is earlier than the

earliest two digit date in the database. Hazama, however, discloses an analogous date processing

system wherein, for a 100 year window, the pivot date for the window is selected based on a two digit

year date that is smaller than the smallest two digit year date in the database (e.g. if smallest two digit

date stored in database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of

translated document, last paragraph). It would have been obvious to one of ordinary skill in the art of

data processing to combine the teachings of the cited references. The ordinary skilled artisan having

read Ohms would immediately see the need to determine which 100 year span to use. This

determination would have led the ordinary skilled artisan to the Hazama reference, which teaches the

pivot date being smaller than the smallest two digit date in the database having all the dates within a

100 year period as a solution to restrict the selection of Ohms's window and thereby forcing all dates

already stored in the database to fall in the 20th century.

62.   Claim 76 is rejected under 35 U.S.C. 103(a) as being unpatentable over Ohms in view of

Hazama, further in view of Booth.

As to claim 76, Ohms substantially discloses the claimed invention. In particular, Ohms

discloses the claimed 'method of processing dates in a database' by presenting a computer -

implemented method for processing date outside the twentieth century (see title, p 244 et seq). Ohms

further discloses the step of 'providing a database with dates stored therein according to a format

wherein MI M2 is the numerical month designator, DI D2 is the numerical day designator, and YI Y2 is

the numerical year designator' by detailing a short Gregorian format (MMDDYY) to represent dates (p

247, see table 1), wherein said dates fall within a 10 decade or a 100 year window (p 249, left-hand

column, lines 3-7 et seq) and wherein the 100 year window contains dates that span in the 21st

century (2000-2024) (p 248, right hand column). Ohms also discloses the step of 'determining a

century designator CI C2 for each date in the database, C1 C2 having a first value if Y1 Y2 is less

than YAYB and having a second value if YI Y2 is equal to or great --r than YAYB' by indicating that

years that are later or equal (25-99) to the pivot date (2_'>) fall within the 20th century thereby

equating CIC2 to 19 (i.e. 1925-1999), whereas dates that are earlier 900-24) than the pivot date fall

within the 21st century thereby equating C1 C2 to 20 (Le. 2000-2024) (see p 248, right-hand column.

Additionally, Ohms discloses the step of reformatting the symbolic representation of each symbolic

representation of a date in the database, without the addition of any new data field to the database,

with the reformatted symbolic representation of each date in the database having the values CI C2,

Y1 Y2, Ioll M2, and DI D2, in order to facilitate further processing of the reformatted symbolic;

representations of each of the symbolic representations of each of the dates' by indicating that upon

determining that a two digit date falls within the 20th or the 21st the century, it is expressed in

accordance with its corresponding century without actually adding a new data field or modifying the

database (i.e. 25-99 -----> 1925-1999, and 00-24 -----> 2000-20:14) (p 248, right hand column).

Regarding the step of selecting a window with a YAYB value for a pivot date of the window, YAYB

being no later than the earliest YI Y2 year designator in the database; Ohms discloses specifying a

year as the desired starting point (pivot date) of the 100 year window (p 248, right hand column, 2nd

paragraph). Ohms does not particularly detail that the pivot date is earlier than the earliest two digit

date in the database. Hazama, however, discloses an analogous date processing system wherein, for

a 100 year window, the pivot date for the window is selected based on a two digit year date that is

smaller than the smallest two digit year date in the database (e.g. if smallest two digit date stored in

database is 73, the pivot date for the 100 year window is chosen to be 72). (page 4 of translated

document, last paragraph). It would have been obvious to one of ordinary skill in the art of data

processing to combine the teachings of the cited references. Hazama's teaching of the pivot date

being smaller than the smallest two digit dare in the database would restrict the selection of Ohms's

window and thereby forcing all dates already stored in the database to fall in the 20th century.

Further, Ohms and Hazama do not specifically, disclose the step of 'sorting the dates in the form CI

C2 YIY2 MIM2 DID2. However, Booth discloses an analogous system that utilizes the Clipper

programming language to process dates stored in a database to thereby derive other dates therefrom

(p 939, lines 1-3 et seq.) In particular, analogously to Ohms and Hazama, Booth discloses the SET

EPOCH command for comparing an entered two digit date with the year digit (pivot date) of the epoch

setting to determine the century to place the date into (p 941, see SET EPOCH paragraph et seq)

thereby converting a six digit date (MMDDYY) into a corresponding eight digit date (C 1 C2Y 1 Y2M 1

M2D 1 D2) (see p. 940941). Additionally, Booth complements Ohms and Hazama by suggesting the

sorting of converted dates after having been reformatted by the SET EPOCH command (p. 945). It

would have been obvious to one of ordinary skill in the art of data processing at the time of the instant

invention to combine the teachings of the cited references. Booth's teaching of sorted reformatted

dates would facilitate the Ohms-Hazama's system to return the reformatted dates in chronological

sequence. And, it would therefore be very useful when indexing the database in date order, as

suggested by Booth in page 945.

Remarks

63.   The Examiner has carefully reviewed Applicant's arguments and Expert's declaration

submitted on October 15, 2003 (Paper # 32). The arguments are not deemed persuasive and

declaration is not effective to overcome the rejections of record.

Response to Applicant's Argument Dated October 15, 2003

64.   The Applicant argued, "the Examiner's complete dismissal of the Declaration of

Toreson, without any rebuttal of the assertions made by Mr. Thoreson as an expert in the

field, is entirely improper". In response to this line of argument, the Examiner points out that

the final rejection of April 16, 2003 indicated that the Expert's Declaration had been carefully

reviewed and that the Expert declaration had substantially reiterated the same arguments

offered in Applicant's remarks. Consequently, a separate response was not deemed

necessary. The Examiner once again maintains the position stated above for the reasons

indicated hereinabove.

65.   Applicant argued, "a failure to consider the reference(s) as a whole, and/or fail to result in a

combination that contains all of the claimed elements, and, therefore fail to support a prima facie case

of obviousness". In response to this line of argument, the Examiner respectfully disagrees with the

preceding argument because all the claims in this application were properly rejected under 35 U.S.C.

103(a). As discussed above, the Examiner relied on the combination of Shaughnessy, Hazama and

Booth to establish the Prima facie case of obviousness in accordance to the Graham V. Deere

requirements. Further, Applicant is reminded that one cannot show nonobviousness by attacking

references individually where the rejections are based on combinations of references. See *In re

Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.,* 800 F.2d 1091, 231

USPQ 375 (Fed. Cir. 1986).

66.   The Applicant also argued, "The Examiner's explanation of the nature of the so-called

"Examiner's rejections" still does not explain why those comments of the Examiner were not in

accordance with the requirements of the M.P.E.P.   The Examiner respectfully submits that the

general comments about Shaughnessy and Hazama are not particularly intended as a rejection of the

claimed invention. They are presented as a way to offer an overview of how the references generally

obviate the claimed invention. Applicant should therefore regard these comments as a prelude to the

35 U.S.C. 103(a) rejections of the claimed invention over the cited references.

67.   Applicant argued that "certain specific in the Examiner's continuing arguments represented in

these paragraphs include, the Examiner's ignoring the Col. 11, line 3 - Col. 18, line 25 explanation of

how Shaughnessy actually works in regard to the Examiner's comments in Paragraph 7". However,

the Examiner disagrees with the preceding argument because the Examiner specifically summarized

the Shaughnessy reference in the paragraph above including the contents of Col. 11, line 3-Col. 18,

line 25. Therefore, the Examiner reiterates that the Shaughnessy reference is an essentially

complete teaching of the claimed subject matter. In particular, it teaches modifying those dates that

have a two-digit identifier less than some predetermined pivot date, changing the format of the date,

and sorting the results. However, Shaughnessy does not explicitly state that the predetermined pivot

date is less than any date in the database.

The Applicant also argued "that Applicant does not argue that Ohms is non-analogous art, to

the extent that it shows a form of windowing, but, rather that Ohms uses windowing different and for

a different purpose, regarding Paragraph 24". The Examiner disagrees with the line of argument that

"Ohms uses windowing different and for a different purpose" because the windowing of Ohms and

the present application is similar because the representation of dates in **the Ohms reference is not**

limited to a Lillian format. In table 1, page 247, Ohms indicates that dates are stored in the

database in Gregorian format to ambiguous convert six digit dates into unambiguous eight digit

dates, thereby resolving the Y2K problem which is similar to the problem that the present application

wants to resolve.

Further, Applicant argued that the 112 rejections presented in last office action should have

been made in the first office action in this merged reissue proceeding to give applicant an

opportunity to address them twice. Applicant's arguments are spurious at best, and have no legal

basis to support the contention that he should have been allowed to address the rejection in the prior

response. The rejections were made in a non-final action. Thus, applicant was not denied an

opportunity to respond to the rejection.

68.   The Applicant further argued that "neither section of the MPEP cited by the Examiner

supports the Examiner's contention that a new matter rejection is proper, as opposed to a rejection

based upon a lack of enabling disclosure or an inadequate description in the specification and

claims as filed being present to support the newly added claims". The Examiner disagrees with the

preceding argument because M.P.E.P 608.04 (a) is clearly states in relevant-part that *Matter not in*

the original specification, claims, or drawings is usually new matter.

Additionally, MPEP 608.04(f) states in relevant-part:

No amendment may introduce new matter in the disclosure of an application

When new matter is introduced into the specification, the amendment should be objected to under

35 U.S. C. 132 (35 U.S. C. 251 if a reissue application) and a requirement made to cancel the new

matter. The subject matter, which is considered to be new matter must be clearly identified by the

examiner. if the new matter has been entered into the claims or affects the scope of the claims, the

claims affected should be rejected under 35 U.S. C. 112, first paragraph, because the new matter is

not described in the application as originally filed. Consequently, the new matter rejection is proper.

69.   Applicant argued that "the accompanying Declaration of Winner, in addition to prior

submissions of applicant, support the fact that one skilled in the art, given the disclosure that the

invention would facilitate a sort of C1C2Y1Y2M1M2D1D2 would realize that sorting of just

C1C2Y1Y2 is possible, without undue experimentation".   In response, the Examiner respectfully

submits that, as discussed in the office action, the claimed C1C2Y1Y2 format results in a faster sort

than the C1C2Y1Y2M1M2D1D2 format described in the specification. Applicant failed to provide

any evidence to support the contention that such disclosed sort would enable one of ordinary skill in

the art to realize the claimed feature faster sort. For the same reasons, it is concluded that the

Applicant was not in possession of the faster sorting scheme at the time of the invention.

70.   Applicant argued that the claims do not have to use the precise language of the specification

and the fact these words are not expressly stated in the specification is not prima facie evidence of

lack of enablement. In response, the Examiner submits that it is unclear as to where in applicant's

specification such adequate description and enablement was provided to support the claimed

language reformatting date "without modifying 'and "without changing" date representations. If by

definition reformatting dates means modifying or changing the format/representation of dates, how

can applicant's claimed limitations achieve such reformatting dates without changing the date

representations. Such language is likely to confuse the ordinary skilled artisan. The Examiner cannot

help but wonders that if such limitations are so material to be included in the claims, why on earth

did Applicant not take the time to include them in the original specification. The fact that the

specification is devoid of the expressions "without changing" or "without modifying" or any

equivalents thereof is prima facie evidence that the specification is non-enabling and that these

newly added limitations in the claims are essentially new matter. If Applicant intends for the "without

changing" or "without modifying" limitations to imply not having to add fields to store the extra two

year **digits after the date** conversion, then the Examiner submits that the cited claims fail to capture

the essence of the invention. Further, the specification does not teach the ordinary skilled artisan

how to achieve such end without undue experimentation, and Applicant's response failed to point

where in the specification adequate support is provided for such interpretation. Therefore, the 112

rejection of claims 33, 60-61, 64-65 and 70 is proper and is hereby sustained.

71.   Applicant argued, "the claims 16-30, 32, 34-67, 69-71, 75 and 76 utilized the terms "pivot

date" and "pivot year" precisely as the art understands them".   However, the Examiner disagrees

with the preceding argument because Applicant simply made a general statement that the terms

"pivot date" and "pivot year" is utilized precisely as the art understands them without explaining,

exactly, for the record how he understands these terms are used in the art. This general allegation

is not sufficient to overcome the rejection under 35 U.S.C. 112. Thus, the Examiner Submits that to

the extent that the use of the cited terms in this application were limited to their well-known definition

in the art, the specification would not need define them. However, the context in which applicant

used the cited terms, as claimed, exceeds the scope of their well-known meaning in the art. Since

the specification does not provide any clarification or support for Applicant's tailored used of those

terms, the 112 rejections are proper.

72.   Applicant argued, "the Declaration of Toreson and corresponding arguments of counsel

sufficiently point out the reasons that the Specification as printed in the Dickens patent alone

supports the claims 36-37, 40-41, 51-59, 69 and 38-39, 42-43". However, It is not clear as which

portion of the specification or to which portion of the Declaration the Applicant is referring to in order

to support such line of argument. It has been noted, throughout this response, that the Applicant is

making many general allegation without specifically pinpointing where in the specification or the

Declaration support for certain arguments can be found.

73.   Applicant argued, "the addition of the Appendix A disclosure by certificate of correction simply

serves to reinforce this disclosure supporting the claim recitations in claims 16-25, 31-33, 66-67, 72,

and 36-43". The Examiner respectfully submits that the certificate of correction does not support the

aforementioned claimed limitations because the certificate of correction simply correct certain

typographical errors in the reference cited section and the abstract. Also, There is no addition of the

Appendix A by certificate of correction. Again the Applicant is making general allegation of supports

for the claims without specifically explained the content of the Appendix A and how specific portions

of the Appendix support the claimed language.

74.   The Applicant argued, "The specification fully and clearly discloses the claimed invention such

that it can be practiced without undue experiment, both as printed in the issued Dickens patent and

with the content of Appendix A". The Examiner, however, disagrees with the preceding argument

because, throughout this response, the Applicant had made many general allegations without

explanations to supports such general allegations. Such allegations on the part of applicant is

improper to satisfy the enablement requirements of 35 U.S.C. 112, according to which the

specification must fully and clearly disclose the invention such that it can be practiced without undue

experiment. In other words, the specification must be self-sufficient. Applicant has failed to meet that

threshold, and has therefore not provided adequate support in the specification for the claimed

limitation.

75.   Applicant argues that one skilled in the art would understand that this ambiguity is related

to any two centuries overlapping a turn-of-the-century date". However, the Examiner submits that

"calling for the avoidance of ambiguity by reformatting or converting date representation" is

improper. Applicant cites to column 1, lines 25-35 of the Dickens patent in support of the claimed

limitation. Thus, to the extent that the claimed "ambiguity" is limited to the inability to discern

between two digit year dates falling in the 19th and 20th centuries, the cited textual portion does

support the claimed limitation.

76.   Applicant concedes that Shaughnessy does suggest selection of a pivot date to determine the

end of a 100-year window. However, Applicant argues that the disclosed Pivot date does not use

the earliest date in the database as recited in the claim invention.   The Examiner respectfully

submits that the office action specifically referred to Shaughnessy's suggestion to use the current

date as the pivot date of the 100 year window, wherein said current date compares low to all other

dates in the database (col. 6, lines 4-5; col. 7, lines 16-17). Therefore, Shaughnessy does suggest

the selection of a pivot date to determine the end date of a 100 year window. Additionally, applicant

is reminded that the office action relied upon Hazama for its unequivocal teaching of `a pivot date

that is earlier than all other dates in the database' to complement Shaughnessy. Therefore, the

claimed limitation of selecting YaYb earlier... was said to be taught by the Shaughnessy-Hazama

combination.

77.   Last, the Applicant argued that "the treatment of, e.g., one or two date data fields at a time in

a subroutine is not the claimed conversion of each of the dates in the database to facilitate further

processing of all of the dates in the database; and that Shaughnessy and Hazama employ the basic

principles of windowing but in other than the claimed processes of the Dickens patent". However,

the Examiner respectfully disagrees with the preceding argument and submits that this argument

was fully addressed above. The Applicant should duly note that the nature of the various solutions

to the Y2k problem involves eliminating the date ambiguities between the 19th and 20th centuries

such that such converted dates can be properly used in various applications. This is implicit any

solution to the Y2k problem. In particular, as discussed in the office action and in the preceding

paragraphs, Shaughnessy's Y2k solution relates to reformatting dates from six digit to 8 digits such

that the reformatted dates can be used in subsequent applications. Applicant went on to assert that

solutions proposed by Hazama and Shaughnessy are different from that of the claimed invention.

This evidence, however, does not support such contention. Applicant is attempting to restrict the

date conversions in Shaughnessy to a single instance when the reference is concerned with

removing ambiguities between dates of two centuries, and while the reference clearly teaches the

conversion of dates. Applicant seems to confuse Shaughnessy's conversion of the dates one at a

time with the so-called conversion of a single date. Such misconception on the part of Applicant

cannot be relied upon to distinguish the claimed invention from the prior art of record.

Response to Declaration of Expert Witness Dated October 15, 2003

78   The Expert's Declaration has carefully reviewed and that the Expert declaration substantially

reiterates that same arguments offered in Applicant's remarks, which were fully addressed in

paragraphs 1-14 above. In Addition, beside that clarification, the Examiner will address the Expert's

declaration in the paragraphs that follows.

79.   It is noted that Mark Winner is declared to be a Senior Systems Software Engineer-Computer

Consulting for the past 25 Years.

80.   It is noted that, Mark Winner declared to have studied United Patent No. 5,806,063; U.S.

Patent 5,630,118; Japanese Application No. 05-027947; and the articles of Ohms, B.G. Ohms,

Computer Processing of Dates Outside the Twentieth Century, IBM Systems. Journal, Volume 25,

Number 2, 1986, pages 244-51. In addition, Mark Winner declared to be familiar with the Clipper 5

operating system as it existed on the date of the filing of the Dickens patent and have specifically

reviewed the portions to which the Examiner in the above referenced Merged proceeding has made

reference. Mark Winner also declared to have studied portions of the prosecution history of the

Dickens patent before it was originally issued as mentioned in this Declaration. Additionally, Mark

Winner declared to have also studied the claims of the Dickens patent as originally issued and those

added in the Reissue application in the above referenced Merged Proceeding; as well as the content

of Exhibit a filed with the dickens patent application.

81.   Mark Winner declared to be based his opinions in this Declaration upon his knowledge of

the art as a person of at least ordinary skill in the art at the time of the filing of the Dickens patent

and on the above referenced materials, which he declared to have reviewed.

82.   Mark Winner declared to understand the Dickens patent in its Specification, with or without

the Exhibit A referred to in the body of the Specification to have disclosed to one of ordinary skill

in the art to which it pertains at the time of its filing.

83.   Mark Winner declared to summarize and note all the elements of the Dickens Patent in

paragraphs 6-14 of the Declaration.

20.   Mark Winner declared to summarize and note all the elements of the Shaughnessy Patent

in paragraphs 18-26 of the Declaration.

84.   Mark Winner declared, there is no teaching or in Shaughnessy of:

       selecting a 10-decade window with a $Y_A Y_B$ value for the first decade of the window, $Y_A Y_B$

       being no later than the earliest $Y_1 Y_2$ year designator in the database; ....

The Shaughnessy method selects a to-decade window utilizing the "date the system was installed."

In response the Examiner respectfully submits that Shaughnessy was not used in the rejection of the

claims for that purpose. Only The Hazama reference was used to compensate for the

aforementioned limitations. Therefore, It would have been obvious to one of ordinary skill in the art of

data processing to combine the teachings of the cited references. The ordinary skilled artisan having

read Shaughnessy would immediately see the need to determine which 100 year span to use. This

determination would have led the ordinary skilled artisan to the Hazama reference, which teaches the

pivot date being smaller than the smallest two digit date in the database having all the dates within a

100 year period as a solution to restrict the selection of Shaughnessy's window and thereby forcing all

dates already stored in the database to fall in the 20th century.

85.   Mark Winner declared, there is no teaching or suggestion in Shaughnessy of:

       determining a century designator C1C2 for each symbolic representation of a date in the

       database, C1C2 having ...; ....

       The teaching of Shaughnessy is to determine a century designator for at most two date

data representations being processed in a called subroutine at any given time. The Examiner

respectfully disagrees because Shaughnessy, in column 7, lines 7-15, particularly discloses the

comparison of each date in the database with the corresponding portion of the end of the 100

year cycle elate. Depending on whether said date is less than or equal to or greater than the end

of the 100 year cycle date, a century value C 1 or C2 is attached to the date, thereby reformatting

the date to a YYYYMMDD format, which corresponds to applicant's claimed

C1C2Y1Y2M1M2D1D2 date format. Shaughnessy particularly refers to conversion of dates not

just a single date in the database as erroneously contended by applicant (see column 6, lines

60-65). Since the disclosed windowing is part of the conversion process for each date, therefore,

said windowing process performed for each date in the database.

86.   Mark Winner declared "there is no teaching or suggestion in Shaughnessy of

reformatting the symbolic representation of the date with the values C1C2Y1Y2M1M2 and D1D2

to facilitate further processing of the dates. The teachings of Shaughnessy is to reformat at most

two dates at a time in the called subroutine and the return to the program from the called

subroutine of an indicator of the result of the processing of the two reformatted date data entries.

Shaughnessy does not teach facilitating "further processing of the dates" by "reformatting the

symbolic representation of the date" "for each symbolic representation of a date in the database".

The Examiner respectfully submits that Shaughnessy teaches the return of a parameter to the

application program, wherein said parameter is indicative of the result of the date operation

having been performed. Applicant should duly note that while Shaughnessy discloses a plurality

of subroutines, which are called and returned to the application program, all the subroutines are

not specifically concerned with date conversion. See generally column 4. Shaughnessy clearly

indicates that the parameter returned to the application program can be either `a *date field* or a

code indicative of a result of the date operation having been performed.' See column 2, lines

37-40. Also, the returned parameter can be a *date field* (i.e. capable of having date entries

therein). Consequently, the return of the date field may contain reformatted dates that can be

used to facilitate further processing as suggested by Shaughnessy in column.2, lines 45-54.

87.   Mark Winner declared to "there is no teaching or suggestion in Shaughnessy of sorting

the symbolic representations of dates;

The method of Saughnessy does not teach "sorting all the "symbolic representation of

dates." It teaches only the comparison of one date to a fixed date or two dates to each other

in the called subroutine and returning to the program an indication of the results of the

comparison". The Examiner respectfully submits that Shaughnessy was not used for showing

"sorting all the "symbolic representation of dates." Shaughnessy and Hazama do not

particularly disclose the additional step of sorting the symbolic representations of dates, after

the step of reformatting. However, Booth discloses an analogous system that utilizes the

Clipper programming language to process dates stored in a database to thereby derive other

dates therefrom (p. 939, lines 1-3 et seq. In particular, analogously to Shaughnessy, Booth

discloses the SET EPOCH command for comparing an entered two digit date with the year

digit (pivot date) of the epoch setting to determine the century to place the date into (p. 941,

see SET EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a

corresponding eight digit date (CIC2YIY2MIM2DID2) (see p. 940-941). Additionally, Booth

complements Shaughnessy and Hazama by suggesting the sorting of converted dates after

having been reformatted by the SET EPOCH command (p. 945). It would have been obvious

to one of ordinary skill in the art of data processing at the time of the instant invention to

combine the teachings of the cited references. Booth's teaching of sorted reformatted dates

would facilitate the Shaughnessy- Hazama's system to return the reformatted dates in

chronological sequence. Such disclosed sorting would therefore be very useful when indexing

the database in date order, as suggested by Booth on page 945.

88.   Mark Winner Declared that "there is no teaching or suggestion in Shaughnessy of:

reformatting each symbolic representation of a date into the format C1C2Y1Y2M1M2D1D2, nor

sorting the symbolic representation of dates and their associated information back into the

database, nor manipulating information in the database having the reformatted date information

therein." The Examiner respectfully submits that it should be apparent from the cited textual

portions of Shaughnessy relied upon above as well as the foregoing discussions, that the

reformatted dates are necessarily already stored in the date field returned by the subroutine,

wherein said date field is forwarded to the application program for further processing.   It is the

Examiner's position that the solution of storing the symbolic representations back in the database

is taught by Shaughnessy as the available and inevitable solution.

89.   Mark winner declared "there is no teaching or suggestion in Shaughnessy of: converting pre-

existing date information [within a database] having a different format into the format wherein M1M2

is the numerical month designator, D1D2 is the numerical day designator and Y1Y2 is the numerical

year designator." The Examiner respectfully submits that Shaughnessy discloses the aforementioned

limitations as the converting the current date in a six digit format (YYMMDD), wherein YY represents

the year, MM represents the month and DD represents the day (col. 8, lines 18-27 et seq).

90.   Mark Winner declared that "there is no teaching or suggestion in Shaughnessy of: selecting YA

YB such YB is 0 (zero)." The Examiner submits that Shaughnessy was not used for that purpose.

Booth was used to show such claimed limitations. Therefore, Shaughnessy and Hazama do not

specifically disclose the step of selecting Y_A Y_B such that Y_B is 0 (zero). However,

Booth discloses an analogous system that utilizes the Clipper programming language to process

dates stored in a database to thereby derive other dates therefrom. See p. 939, lines 1-3 et seq. In

particular, analogously to Shaughnessy, Booth discloses the SET EPOCH command for comparing

an entered two digit date with the year digit (pivot date) of the epoch setting to determine the century

to place the date into (p. 941, see SET EPOCH paragraph et seq) thereby converting a six digit date

(MMDDYY) into a corresponding eight digit date (C 1 C2Y 1 Y2 M 1 M2D 1 D2) (see p. 940-941).

Additionally, Booth complements Shaughnessy and Hazama by suggesting that the pivot date be set

to 90 by selecting set epoch to be 1990 (Le. YAYB = 90), such that YB equals to zero (p. 942). It

would have been obvious to one of ordinary skill in the art of data processing at the time of the instant

invention to combine the teachings of the cited references. Booth's teaching of setting the pivot date

to a predetermined value would enable users of the Shaughnessy- Hazama's system to have direct

control over the return of reformatted dates to thereby preset the date processing system in

accordance with their needs. 11. Claims 11-15 are rejected under 35 U. S. C. 103 (a) as being

unpatentable over Shaughnessy in view of Hazama, and further in view of Booth.

91.   Mark Winner declared that "There is no teaching or suggestion in Hazama nor in the in the

combination of Shaugnessy and Hazama of:

    Determining a century designator C1C2 for each symbolic representation of a data in the

database C1C2 having .....

The teaching of Hazama, or Shaughnessy in view of Hazama, is to determine a century designator for at most date data representations being processed in a called subroutine/module at any given time." However, the Examiner respectfully disagrees because Shaughnessy, in column 7, lines 7-15, particularly discloses the comparison of each date in the database with the corresponding portion of the end of the 100 year cycle date. Depending on whether said date is less than or equal to or greater than the end of the 100 year cycle date, a century value C 1 or C2 is attached to the date, thereby reformatting the date to a YYYYMMDD format, which corresponds to applicant's claimed C1C2Y1Y2M1M2D1D2 date format. Shaughnessy particularly refers to conversion of <u>dates</u> not just a single date in the database as erroneously contended by Declarant (see column 6, lines 60-65). Since the disclosed windowing is part of the conversion process for each date, therefore, said windowing process performed for each date in the database.

92.   Mark Winner declared "there is no teaching or suggestion in Hazama or Shaughnessy in view of Hazama of reformatting the symbolic representation of the date with the values C1C2Y1Y2M1M2 and D1D2 to facilitate further processing of the dates. Hazama was not used for showing the aforementioned claimed featured. Therefore, the Examiner submits that the teachings of Shaughnessy is to reformat at most two dates at a time in the called subroutine and return to the program from the called subroutine of an indicator of the result of the processing of the two reformatted date data entries. Shaughnessy does not teach facilitating "further processing of the dates" by "reformatting the symbolic representation of the date" "for each symbolic representation of a date in the database".

The Examiner respectfully submits that Shaughnessy teaches the return of a parameter to the application program, wherein said parameter is indicative of the result of the date operation having

been performed. Declarant should duly note that while Shaughnessy discloses a plurality of

subroutines, which are called and returned to the application program, all the subroutines are not

specifically concerned with date conversion. See generally column 4. Shaughnessy clearly

indicates that the parameter returned to the application program can be either a *date field* or a

code indicative of a result of the date operation having been performed. See column 2, lines 37-40.

Also, the returned parameter can be a *date field* (i.e. capable of having date entries therein).

Consequently, the return of the date field may contain reformatted dates that can be used to

facilitate further processing as suggested by Shaughnessy in column 2, lines 45-54.

93.   Mark Winner declared that "there is also no teaching or suggestion in the combination of

Shaughnessy and Hazama as applied by the Examiner of: Sorting the symbolic representations of

dates." The Examiner respectfully submits that neither Shaughnessy nor Hazama was used for

showing the aforementioned limitations. Only Booth was used for that purpose. Therefore,

The Examiner respectfully submits Shaughnessy and Hazama do not particularly disclose

the additional step of sorting the symbolic representations of dates, after the step of

reformatting. However, Booth discloses an analogous system that utilizes the Clipper

programming language to process dates stored in a database to thereby derive other dates

therefrom (p. 939, lines 1-3 et seq. In particular, analogously to Shaughnessy, Booth discloses

the SET EPOCH command for comparing an entered two digit date with the year digit (pivot

date) of the epoch setting to determine the century to place the date into (p. 941, see SET

EPOCH paragraph et seq) thereby converting a six digit date (MMDDYY) into a corresponding

eight digit date (CIC2YIY2MIM2DID2) (see p. 940-941). Additionally, Booth complements

Shaughnessy and Hazama by suggesting the sorting of converted dates after having been

reformatted by the SET EPOCH command (p. 945). It would have been obvious to one of

ordinary skill in the art of data processing at the time of the instant invention to combine the

teachings of the cited references. Booth's teaching of sorted reformatted dates would facilitate

the Shaughnessy- Hazama's system to return the reformatted dates in chronological

sequence. Such disclosed sorting would therefore be very useful when indexing the database

in date order, as suggested by Booth on page 945.

94.   Mark Winner Declared that "there is no teaching or suggestion in Shaughnessy of:

reformatting each symbolic representation of a date into the format C1C2Y1Y2M1M2D1D2, nor

sorting the symbolic representation of dates and their associated information back into the

database, nor manipulating information in the database having the reformatted date information

therein." The Examiner respectfully submits that it should be apparent from the cited textual

portions of Shaughnessy relied upon above as well as the foregoing discussions, that the

reformatted dates are necessarily already stored in the date field returned by the subroutine,

wherein said date field is forwarded to the application program for further processing.   It is the

Examiner's position that the solution of storing the symbolic representations back in the database

is taught by Shaughnessy as the available and inevitable solution.

95.   Mark winner declared "there is no teaching or suggestion of: converting pre-existing date

information [within a database] having a different format into the format wherein M1M2 is the

numerical month designator, D1D2 is the numerical day designator and Y1Y2 is the numerical year

designator." The Examiner respectfully submits that Shaughnessy discloses the aforementioned

Application/Control Number ●9/512,592; 90/005,592; 90/005,62●0/005727;   Page
90/006,541   202
Art Unit: 2161

limitations as converting the current date in a six-digit format (YYMMDD), wherein YY represents the

year, MM represents the month and DD represents the day (col. 8, lines 18-27 et seq).

96.   Mark Winner declared that "there is no teaching or suggestion of: selecting YA YB such YB is

0 (zero)." The Examiner submits that neither Hazama nor Shaughnessy was used for that purpose.

Booth was used to show such claimed limitations. Therefore, the Examiner respectfully submits that

Shaughnessy and Hazama do not specifically disclose the step of selecting $Y_{sub}.A$ $Y_{sub}.B$ such

that $Y_{sub}.B$ is 0 (zero). However, Booth discloses an analogous system that utilizes the Clipper

programming language to process dates stored in a database to thereby derive other dates

therefrom. See p. 939, lines 1-3 et seq. In particular, analogously to Shaughnessy, Booth discloses

the SET EPOCH command for comparing an entered two digit date with the year digit (pivot date) of

the epoch setting to determine the century to place the date into (p. 941, see SET EPOCH paragraph

et seq) thereby converting a six digit date (MMDDYY) into a corresponding eight digit date (C 1 C2Y 1

Y2 M 1 M2D 1 D2) (see p. 940-941). Additionally, Booth complements Shaughnessy and Hazama by

suggesting that the pivot date be set to 90 by selecting set epoch to be 1990 (Le. YAYB = 90), such

that YB equals to zero (p. 942). It would have been obvious to one of ordinary skill in the art of data

processing at the time of the instant invention to combine the teachings of the cited references.

Booth's teaching of setting the pivot date to a predetermined value would enable users of the

Shaughnessy- Hazama's system to have direct control over the return of reformatted dates to thereby

preset the date processing system in accordance with their needs. Thus, Claims 11-15 were properly

rejected under 35 U. S. C. 103 (a) as being unpatentable over Shaughnessy in view of Hazama, and

further in view of Booth.

97.   In paragraph 56-68 of the Declaration, Mark Winner declared that Booth is

non-analogous art since it allegedly does not seek to solve the Y2k problem. Applicant's

interpretation of the Booth reference is simply erroneous. It has been held that a prior art

reference must either be in the field of applicant's endeavor or, if not, then be reasonably pertinent

to the particular problem with which the applicant was concerned, in order to be relied upon as a

basis for rejection of the claimed invention. See In re Oetiker, 977 F.2d 1443, 24 USPQ2d 1443

(Fed. Cir. 1992). In this case, the textual portions of the Booth reference relied upon in the

rejection of the cited claims are particularly concerned with converting two digit year dates to four

digit year dates (see e.g. SET EPOCH, P 941). Booth states in relevant-part:

> "When a two digit year is entered into a date, its year digits are compared with
>
> the year digits of the epoch setting to determine the century to place the date
>
> into. If the two digits are prior to the setting of the SET EPOCH, the year is
>
> assumed to be in the next century. If the digits are greater than or equal to
>
> the SET EPOCH setting, the year is assumed to be in the current century."

Clearly, the foregoing portion of Booth is concerned with solving the Y2k problem in the

manner proposed by Applicant, Shaughnessy and Hazama. That is, Booth's solution is intended to

resolve any possible confusion that could result between dates falling between the 20`h and the

215` centuries. Therefore, the Booth reference is within the field of applicant's endeavor since it

sought to solve the same problem with which applicant was concerned.

98.   Applicant admits that Booth does teach windowing using a ten-decade window, but
alleges that said window is not offered in the context of the claimed invention (e.g. not for facilitating
further processing of dates). Applicant further alleges that Booth does not teach reformatting and
sorting the dates. However, Applicant fails to address the particular portions of Booth relied upon for
the cited limitations. The office action made it clear that Booth teaches the conversion of two digit
year dates to four digit year dates, whereby said four digit year dates are sorted. Applicant's failure
to particularly detail how the claimed limitations are distinguished over the cited portions of Booth is
not in conformance with 37 CFR 111 (b) and the controlling case law. In consequence, Applicant
failed to successfully rebut the prima facie case. The rejection of claims 4, 6 and 8 is hereby
sustained.

99.   Regarding the rejection of claims 11-15, Applicant defers to the arguments presented
for claims 4, 6 and 8 to allege that claims 11- 15 are patentable over the prior art of record. The
Examiner submits that, similarly, the arguments offered in the preceding paragraphs clearly show
that a prima facie case was properly established and that applicant failed to successfully rebut it.
Consequently, the rejection of claims 11-15 is hereby sustained.

100.   Regarding the rejection of claims 16-18, 20, 22, 24-25, applicant defers to the
arguments offered for claim 1 to allege that claims 16-18, 20, 22, 24-25 are patentable over the prior
art of record. The Examiner submits that, similarly, the arguments offered in the preceding
paragraphs clearly show that a prima facie case was properly established and that applicant failed to
successfully rebut it. Consequently, the rejection of claims 16-18, 20, 22, 24-25 is hereby sustained.

101.   Regarding the rejection of claims 19, 21 and 23, applicant defers to the arguments

offered for claims 4,6 and 8 to allege that claims 19, 21 and 23 are patentable over the prior art of

record. The Examiner submits that, similarly, the arguments offered in the preceding paragraphs

clearly show that a prima facie case was properly established and that applicant failed to successfully

rebut it. Consequently, the rejection of claims 19, 21 and 23 is hereby sustained.

102.   Regarding the rejection of claims 26-30, applicant defers to the arguments offered for

claims 11-15 to allege that claims 26-30 are patentable over the prior art of record. The Examiner

submits that, similarly, the arguments offered in the preceding paragraphs clearly show that a prima

facie case was properly established and that applicant failed to successfully rebut it. Consequently,

the rejection of claims 26-30 is hereby sustained.

103.   Regarding the rejection of claims 31-33, applicant defers to the arguments offered

for claims 1, 11 and 16 to allege that claims 31-33 are patentable over the prior art of record. The

Examiner submits that, similarly, the arguments offered in the preceding paragraphs clearly show

that a prima facie case was properly established and that applicant failed to successfully rebut it.

Consequently, the rejection of claims 31-33 is hereby sustained.

104.   Regarding the rejection of claims 34-59, applicant defers to the arguments

offered for claims 1 and 16 to allege that claims 34-59 are patentable over the prior art of record.

The Examiner submits that, similarly, the arguments offered. in the preceding paragraphs clearly

show that a prima facie case was properly established and that applicant failed to successfully rebut

it. Consequently, the rejection of claims 34-59 is hereby sustained.

105.   Regarding the rejection of claims 60-71, applicant defers to the arguments offered for claims I and 4 to allege that claims 60-71 are patentable over the prior art of record. The Examiner submits that, similarly, the arguments offered in the preceding paragraphs clearly show that a prima facie case was properly established and that applicant failed to successfully rebut it. Consequently, the rejection of claims 60-71 is hereby sustained.

106.   Regarding the rejection of claims 72, 73 and 75, applicant defers to the arguments offered for claims 4, 6, 13 and 14 to allege that claims 72, 73 and 75 are patentable over the prior art of record. The Examiner submits that, similarly, the arguments offered in the preceding paragraphs clearly show that a prima facie case was properly established and that applicant failed to successfully rebut it. Consequently, the rejection of claims 72, 73 and 75 is hereby sustained.

107.   Regarding the rejection of claims 74 and 76, applicant defers to the arguments offered for claims 1 and 4 to allege that claims 74 and 7 6 are patentable over the prior art of record. The Examiner submits that, similarly, the arguments offered in the preceding paragraphs clearly show that a prima facie case was properly established and that applicant failed to successfully rebut it. Consequently, the rejection of claims 74 and 76 is hereby sustained.

108.   In paragraphs 69-160 of the Declaration, Mark Winner declared that Ohms in view of Hazama, and as to some claims in further view of Booth would not have made the claimed invention as recited in claim 1 obvious to one of ordinary skill in the art at the time the invention was made, under 35 U.S. 103, Applicant defers to the arguments presented in the previous response,

paper no. 20. In the last office action, the Examiner fully addressed applicant's arguments and

sustained the rejection of these claims over Ohms, Hazama and Booth (See paper no. 22). In

addition, Applicant is now arguing that Ohms' disclosure of a method for processing dates outside

the twentieth century is not equivalent to the claimed `method of processing symbolic

representations of dates stored in a database' because the dates in Ohms' database are allegedly

represented in Lillian format, and poses no Y2K ambiguity. In response to the preceding arguments,

the Examiner respectfully submits that as pointed out in the remarks section of the last office action,

the representation of dates in **the Ohms reference is not limited to** a Lillian format. In table 1,

page 247, Ohms indicates that dates are stored in the database in Gregorian format to ambiguous

convert six digit dates into unambiguous eight digit dates, thereby resolving the Y2K problem.

109. Applicant also argues that Ohms does not disclose the storing of dates to be within a

ten decade window because it suggests a variety of date ranges. Applicant also argues that Ohms

does not disclose selecting the ten-decade window based upon dates actually stored in the

database. This reading of the Ohms reference is inaccurate. The fact that Ohms discloses a variety

of date ranges including the ten-decade window does not imply that it teaches away from using the

ten a particular date range. In fact, Ohms specifically proposes the ten-decade window as an

excellent choice for the date conversion. See page, 248, right hand column, and 2nd paragraph.

Further, Ohms discusses at page 249, in the paragraph entitled "Computational Considerations" the

storage requirements for maintaining dates in the different formats in the database. Consequently,

Ohms does disclose the cited limitations.

110.   Applicant generally reiterates that Ohms is non-analogous art since it allegedly does

not seek to solve the Y2k problem since it stores dates in a Lillian format. Applicant's interpretation

of the Ohms reference is simply erroneous. It has been held that a prior art reference must either

be in the field of applicant's endeavor or, if not, then be reasonably pertinent to the particular

problem with which the applicant was concerned, in order to be relied upon as a basis for rejection

of the claimed invention. See In re Oetiker, 977 F.2d **1443, 24 USPQ2d 1443 (Fed.** Cir. 1992). In

this case, the textual portions of the Ohms reference relied upon in the rejection of the cited claims

and in the preceding remarks are particularly concerned with converting two digit year dates to four

digit year dates (see e.g. table 1, page 247) where Ohms teaches the conversion of a short

Gregorian format (YYMNMD) date into a Gregorian date format (MMDDYYYY). Further, at page

248, last paragraph of the first column, Ohms states in relevant-part:

"End users usually enter two digits for the year in a date and

understand the ambiguity that this represents... entry of all four

digits may be required."

Clearly, the foregoing portions of Ohms are concerned with solving the Y2k problem in the

manner proposed by Applicant and Hazama. That is. Ohms' solution is intended to resolve any

possible confusion that could result between dates filling between the 20th and the 21St centuries.

Therefore, the Ohms reference is within the field of applicant's endeavor since it sought to solve

the same problem with which applicant was concerned.

111.   Applicant defers to the arguments presented in the prior response to rebut the rejection of claims 16-76 over Ohms in view of Hazama and further in view of Booth for some claims. The Examiner submits that a prima facie case was properly established in the last office action, and Applicant's arguments were traversed in the Remarks section of the office action. Consequently, the Examiner's remarks regarding applicant's arguments are herein incorporated by reference, and the rejections of claims 1-76 over the Ohms, Hazama and Booth combination are proper and are hereby sustained.

112.   Mark Winner declared that Shaughnessy does not disclose the claimed step of selecting YaYb no later than the earliest year designator in the database. Applicant cites to column 6, lines 6-22 to submit that Shaughnessy only teaches the end of the 100 year cycle without regard for the earliest date in the database. Applicant's reliance on the cited textual portion of Shaughnessy to draw such conclusion is erroneous and misleading. The textual portion cited by applicant essentially indicates that the end of the 100-year window is determined based on the current date (lines 15 22). Further, the office action specifically referred to Shaughnessy's suggestion to use the current date as the pivot date of the 100 year window, wherein said current date compares low to all other dates in the database (col. 6, lines 4-5; col. 7, lines 16-17). Therefore, Shaughnessy does suggest the selection of a pivot date to determine the end date of a 100 year window. Additionally, applicant is reminded that the office action relied upon Hazama for its unequivocal teaching of `a pivot date that is earlier

than all other dates in the database' to complement Shaughnessy. Therefore, the claimed limitation of selecting YaYb earlier... was said to be taught by the Shaughnessy-Hazama combination.

113. Mark Winner declared that the Shaughnessy-Hazama combination does not teach the

determination of a century designator for each date in the database. First, applicant asserts that

neither reference expressly teaches the step on "each date representation. This argument was fully

addressed above. Second, applicant argues that even if such teaching is implicit in the cited

references, it would not mean that the combined references teach that all dates in the database are

converted to facilitate further processing. This is a non-sequitur. The   nature of the various

solutions to the Y2k problem involves eliminating the date ambiguities   between the 19`h and

20'h centuries such that such converted dates can be properly used in various applications. This is

implicit any solution **to the; Y2k problem. In particular,** as discussed in the office action and in the

preceding paragraphs, Shaughnessy's Y2k solution relates to reformatting dates from six digit to 8

digits such that the reformatted dates can be used in subsequent applications. Applicant went on to

assert that solutions proposed by Hazama and Shaughnessy are different from that of the claimed

invention. This evidence, however, does not support such contention. Applicant is attempting to

restrict the date conversions in Shaughnessy to a single instance when the reference is concerned

with removing ambiguities between dates of two centuries, and while the reference clearly teaches

the conversion of dates. Applicant seems to confuse Shaughnessy's conversion of the dates one at

a time with the so-called conversion of a single date. Such misconception on the part of Applicant

cannot be relied upon to distinguish the

claimed invention from the prior art of record.

114.   Mark Winner's remaining declaration has already been fully addressed in the preceding

paragraphs. The Examiner reiterates that the rejections of record are proper.

115.   The Expert's declaration substantially reiterates the same arguments offered in

Applicant's remarks, which were fully addressed in paragraphs 1-41 above. Consequently, a

separate response is not deemed necessary.

Oath /Declaration

117.   The Reissue oath/declaration filed with this application is defective because it fails to identify

at least one error, which is relied upon to support the reissue application. Further, it fails to refer to

the amendment of 01/05/2002. See 37 C.F.R.1.175(a) (1), 37 C.F.R. 1.63 (b)(2), and MPEP 1414.

118.   Changes made to the certificate of correction have not been incorporated into the specification

of the reissue application. Applicant is required to submit a

substitute specification, which complies with reissue practice.

119.   Claims 1-76 are rejected as being based upon a defective reissue declaration under 35 U. S.

C. 251 as set forth above. See 37 CFR 1.175. The nature of the defect(s) in the declaration is set

forth in paragraphs above.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner

should be directed to Frantz Coby whose telephone number is 571 272 4017. The examiner can

normally be reached on Maxi-Flex (Monday-Saturday).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor,

Safet Metjahic can be reached on 571 272 4023. The fax phone number for the organization where

this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application

Information Retrieval (PAIR) system. Status information for published applications may be obtained

from either Private PAIR or Public PAIR. Status information for unpublished applications is available

through Private PAIR only. For more information about the PAIR system, see http://pair-

direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the

Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Frantz Coby
Primary Examiner
Art Unit 2171

September 30, 2004